argo data management

Argo data management

23 October 2003

ar-um-02-02

Comments on USER’S MANUAL

ARGO part of the integrated global observation strategy

Argo data management

Comments on User’s manual

1. INTRODUCTION 4

2. COMMENTS ON USER'S MANUAL VERSION 0.9 4

3. COMMENTS ON USERS'S MANUAL VERSION 0.9B 10

4. COMMENTS ON USER'S MANUAL VERSION 0.9C 24

5. COMMENTS ON USER'S MANUAL VERSION 1.0 26

6. COMMENTS ON USER'S MANUAL VERSION 1.0C 74

7. COMMENTS ON USER'S MANUAL VERSION 1.0D 85

1. Introduction

This document contains the comments received on the different versions of the argo data users's manual.

The reader should be aware that the comments apply to two separate issues :

- comments on the users's manual itself,
- comments on the argo data formats.

2. Comments on user's manual version 0.9

From : Roger Goldsmith

Date : 14/12/2001

```
Some minor observation on release 2.0 of Argo formats.
I apologize if some seem trivial but if there is any confusion someone
will get tripped up eventually.

ARGO PROFILE FORMAT
  REFERENCE_DATE_TIME    The DATE_TIME is 14 characters, the convention
                         shows 16 with the inclusion of the 24 and the
                         commented example shows 12, lacking some zeros.
TC : OK

  Q_PROFILE_PSAL_QC      The leading "Q_" appears to be inconsistent
with

                         the manner in which the other QC are defined.
TC : OK


ARGO TECHNICAL DATA FORMAT
  DATA_TYPE     It is defined here as STRING32, but defined as STRING16
                in the profile and trajectory formats.
TC : string16 is ok

  ARGO_HANDBOOK_VERSION    Is this really different than the
HANDBOOK_VERSION

                           defined for the profile and trajectory
formats?
TC : we use HAND_BOOK_VERSION (it can be applied to an other program)


ARGO META DATA FORMAT
  PTT     Here lower case Argos, Orbcomm is used; then uppercase in the
          parameter TRANS_SYSTEM.  What are the conventions that will
allow
          uniform searches?
TC : we should use uppercas

  TRANS_SYSTEM     Here uppercase ARGOS and ORBCOMM are used.  And while
not

                   technically a transmission system, how would
GLOBALGRAM
                   be denoted.
  PLATFORM_SERIAL_NO     Appropriately noted as char type so the use of
"NO"

                         and "number" may cause confusion.  Perhaps "ID"
and
                         "identifier".
TC : yes but we have to wait for the next version of format

  DIRECTION     Here has convention for "A" and "B" while in the profile
format

                it has "A" and "D".  Perhaps all three cases "A", "D"
and "B"
                should be declared in each format.
TC : a descending only float looks strange
```

```
  START_DATE_QC      long_name has "Quality on launch..." and should be
                     "Quality on start ..." of "first descent"?
TC : OK

  SENSOR_SERIAL_NO      Same comment as for PLATFORM use of number versus
                        identifier.
TC : yes but we have to wait for the next version of format

  REPETITION_RATE      The _FillValue should perhaps be just "99999" to
match
                       the type "int".
TC : OK

  TECHNICAL_PARAMETER_NAME      Type should be lowercase char?
  TECHNICAL_PARAMETER_VALUE      Type should be lowercase char?
TC : yes

In various places there are defined:
STATION_PARAMETERS (profile),
PARAMETER (profile, meta),
MEASURED_PARAMETERS (trajectory),
TECHNICAL_PARAMETER_NAME/VALUE (technical, meta)

If these are all different entities then perhaps they should have more
distinctive identifiers.
```

From : Claudia Schmid

Date : 24/01/2002

Thierry,

the document looks good.

Now some comments:

Your question on p.6:
I would use instrument (or profiling float) instead of station. I associate a station with the stopping of a ship at one particular location to obtain one (or many) observation(s), often with different instruments.

'everywhere': replace HH24 with HH (reason: we want only two numbers as hours). In definition of DATE_TIME we should specify: HH: hour of the day (0-23).
TC : OK

`everywhere': `Universal time' is officially `Universal Time Coordinate' and is abbreviated with UTC.
TC : OK

`everywhere': replace Exemple(s) with Example(s)
TC : OK

`everywhere': We were told not to treat the Q as a part of the WMO number. While this is not important now it will be important when we start using BUFR (I will forward the respective email to you).
TC : do we all agree ?

p.8: attributes (with s)

Do we need a reference table for DATA_TYPE?
TC : yes we do.

Do we want to discern between ARGO and non-ARGO profiling floats (AOML has several).

TC : do we or don't we ?

p.9: move PLATFORM_NUMBER to beginning of general information (also for other file formats)

p.9: ... it perform an ascending (perform needs a `s')
TC : OK

CYCLE_NUMBER: Do I understand it correct? I think what you mean is that an ascending/descending profiler has the numbers 1 1 2 2 3 3 ...
It may be good to give two examples.

p.10:
CYCLE_NUMBER: Will we ever have a 0 here? My guess is not, because you defined a cycle as descending, submerged drift, ascending, surface drift (as we also do it). I.e. cycle 0 will only consist of a surface drift.
TC : cycle 0 is the first cycle, it usually has a smaller subsurface drift.

DC_REFERENCE: Unique identifier of the profile ...
I suggest to use instrument (or profiler), since the file is designed to contain many profiles.
TC : DC reference is about a profile. One file may contain profiles from different instruments.

INST_REFERENCE: Why not INST_TYPE (consistent with long name)?
Do we need a reference table or some rules here?

p.11
JULD_LOCATION: ... is generally estimated after ... profile can be different.
TC : OK

LATITUDE/LONGITUDE: remove value
TC : OK

POSITIONING SYSTEM: GPS exists with different decoding systems. Do we want to discern between them here?
TC : do we or don't we ?

PROFILE_PARAM_QC: replace TEMP (temperature) with PARAM (parameter) in description. Similarly on p.13 (PSAL).
TC : OK

p.16: Where can we learn how the HISTORY_QCTEST values are generated? reference necessary.
TC : OK

p.17:
N_MEASUREMENT: columns not properly aligned
N_CYCLE: depends on data set
TC : OK

p.22
POSITION_ACCURACY: fill value can not be 0, because 0 is a valid ARGOS location class. Suggestion: -99
TC : I removed the value

What do we do with GPS position accuracies (also in reference tables appendix)?
Maybe we should give the accuracy in meters (or meter ranges)?

TC : OK, the range has to be defined

POSITION_QC: Here we can use the fill value 0 (consistent with the QC flags in the reference table).
TC : OK

PRES: change definition.
Pressure measurements during surface/submerged drift.
Remove the 2nd and third sentence.
What happens if a float records a negative pressure at the surface? We just store them as we get them (they may later be used to correct drifts of the pressure sensor). I.e. valid_min=0 may be problematic.
TC : maybe should we remove valid_min and valid_max

p.30
TRANS_SYSTEM: IRIDIUM is another possibility.
TC : OK

bottom of page: something is wrong with the table here (on my printout half a line, cut in the verticel, is visible)

p.32
DEPLOY_AVAILABLE_PROFILE_ID: DEPLOY_PROFILE_ID sounds better to me. I would allow "XBT 5, CTD 201" instead of just numbers (Who will remeber what kind of a profile it was after 5 years?)
TC : the string256 allows these names

p.33
SENSOR_MODEL: Conductivity or salinity? I assume that the salinity is estimated on the float computer (and would thus serve as a bed example), but I don't know if that's correct. If you change it we should use the SI units dS/m (deci-Siemens per meter). The sensor resolutions will then change as well.

Concerning the data resolution: This may be affected by the way the floats store the data. Do we need a field for this (like STORAGE_RESOLUTION)?
TC : I have no opinion

WMO table 1770: Do we want to append it here, or just give the web adress? I think the latter may make sense, because the table may change more often than the handbook.
TC : I inserted both.

From : Claudia Schmid

Date : 25/01/02

p.10:
CYCLE_NUMBER: Will we ever have a 0 here? My guess is not, because you defined a cycle as descending, submerged drift, ascending, surface drift (as we also do it). I.e. cycle 0 will only consist of a surface drift.

TC : cycle 0 is the first cycle, it usually has a smaller subsurface drift.

I don't understand your response. Some of our floats transmit data after deployment until they sink for the first time. The associated data is what we put into cycle 0. Therefore cycle 0 (for us) does not include any subsurface drift. Cycle 1 contains an ascending (and if applicable a

descending) profile. The descending profile of a cycle, if it exists, was
obtained before the ascending profile. If I understand you correctly we
use different definitions for cycle 0. However, I'm not sure how your
definition is. Can you clarify this?

Claudia

PS: another small suggestion. I'm a bit challenged with word documents
here. We have star office on unix which can sort of read them, but it
screws up the formatting. So maybe if you add your replies to comments you
can do it in ascii? For the printing of the handbook itself I don't mind
going to a PC.

From : Roger Goldsmith

Date : 06/02/02

Thierry Carval,
some edit for the manual.

1) There still doesn't seem to be a consensus on whether it's
POSITIONING or POSITIONNING
It is used both ways throughout the document, sometimes one way
in the variable and another in the attribute.

TC : according to harrap's, we have to use "positioning" instead of "postionning"

2) p. 32 PTT: ORBCOMM should maybe have two MM throughout?

TC : corrected

3) p. 33 LAUNCH_QC has attributes LAUNCH_DATE_QC

TC : corrected

4) p. 33 START_DATE_QC has a long name of launch date.

TC : corrected

5) p. 27 JULD_START_TRANSMISSION
the attributes have JULD_DESCENT_END
the example has DATE_TIME format rathe than the double type.

TC : corrected

From : Yasushi Talatsuki

Date : 22/02/02

[1] I found that 'NOMINAL_PROFILE_PRES', 'NOMINAL_PARKING_PRES'
and 'PARKING_PRES_QC' field were deleted in format version 2.0,
that were existed in version 1.2b.
I think these information are valuable especially for the floats
that may change parking/profiling pressure on each cycle.
Therefore, I propose to adding following definitions in
the Cycle data block;

```
int NOMINAL_PROFILE_PRES(N_CYCLE);
    NOMINAL_PROFILE_PRES:Comment ="Nominal pressure of the ascending profile for this
cycle";
    NOMINAL_PROFILE_PRES:_FillValue = 99999.f ;
    NOMINAL_PROFILE_PRES:units = "decibar";
int NOMINAL_PARKING_PRES(N_CYCLE);
    NOMINAL_PARKING_PRES:Comment ="Pressure of the submarged drift for this cycle";
    NOMINAL_PARKING_PRES:_FillValue = 99999.f ;
    NOMINAL_PARKING_PRES:units = "decibar";
char NOMINAL_PARKING_PRES_QC(N_CYCLE);
    NOMINAL_PARKING_PRES_QC:Comment = "Quality on parking pressure ofthe cycle";
    NOMINAL_PARKING_PRES_QC:Convensions="Q where Q=[0-9]";
    NOMINAL_PARKING_PRES_QC:_FillValue = "0";
```

TC : nominal values are stored in meta-data. These are not measured values that change at every cycle. They are set once before launch and recorded in meta-data.

[2] In the Profile format, we define "PSAL_CORRECTED_ERROR" to store
estimated error for the salinity correction, but no definition
in the trajectory format. Is it unnessesary for the trajectory format?

TC : if this parameter becomes necessary for trajectory, we shall add it when necessary.

[3] Should we treat the measurement field (PRES, TEMP, PSAL, CNDC) for
the floats that do not measure during the surface/subsurface drift?
(Many APEXs measure only profiles and have no drift data.)

1) Remain all the definition as same in profile format for the float
and filled with _FillValue except for PRES? (PRES may be set to 0.)
2) Only defined PRES/PRES_CORRECTED... and filled with 0 for PRES?
3) Do not define PRES/TEMP/PSAL/CNDC and N_PARAM,TRAJECTORY_PARAMETERS
for that float?

TC : the format contains the measured parameters. If a parameter is not measured, it is not defined in the file.

What do you think about above things?

From : Yasushi Takatsuki

Date : 01/02/02

I should add another thing.

According to the GTSPP code table
(http://www.meds-sdmm.dfo-mpo.gc.ca/meds/databases/ocean/gtsppcodes_e.htm),
4.4 in page 44 should be
GE: BSH (Germany)
JA: JMA (Japan).
TC : ok, done

From : Yasushi Takatsuki

Date : 31/01/02

Does Cycle 0 mean incomplete cycle (the time for the cycle may be shorter or longer than the preprogrammed cycle period) just after launch?
Therefore, in case of APEX, their cycle begin with 1 not 0, because they complete their cycle from just first descent?

The unit of conductivity in GF3 tables presented in 4.3 is "mhos/m", not "mmho/cm". Also, The unit of practical salinity is "---", not "psu".
If we use the units as listed in GF3 tables, we should change the units and "xx_format" for the parameter CNDC and PSAL. (1 mhos/s = 10 mmho/cm)
TC : in our data bases, conductivity is stored in mmho/cm . Do we change it to mhos/s ? I prefer to avoid it.

Does JULD represent either of JULD_ASCENT_END or JULD_START_TRANSMISSION in the Argo trajectory format for ascent profile?
TC : JULD is the data and time of the profile. It is the date and time at the top (surface) of the profile,

wether ascending or descending.

Page 18 : in GTSPP code, letter x in "QxP$" and "QxF$" depends on the institute where QC is performed or failed.
(see http://www.meds-sdmm.dfo-mpo.gc.ca/meds/Databases/OCEAN/GTSPPcodes_f.htm)
For example, QCP$ and QCF$ for MEDS, QAP$ and QAF$ for AOML, and so on. In Argo data format, the information of institution which performed is already presented by the field "HISTORY_INSTITUTION".
Should we use "QCP$" and "QCF$" for Argo data at all institute where QC is performed?

If we use "QCP$" and "QCF$", we must write its definition in the table in 4.7.

Page 20 : N_PARAM should be 2, 3 or 4 because we cannot explain the pressure sensor information in Float sensor information section of the meta-data format if we set N_PARAM is 1, 2 or 3.

Page 27 : Does JULD_START_TRANSMISSION represent the date/time when the float begin to transmit data even if the satellites do not receive it? Or the date/time of the first message which the satellites is received ? I intended former time. Is it right?
TC :I think it represents the date/time when the float begin to transmit data

Page 35 : I remember that "SENSOR_MODEL" associated with sensor model name (example, SBE-41 or FSI EXCELL), so we discussed that we use "SENSOR_TYPE" , which associated with sensor type such as "inductive sensor" or "thermister", rather than "SENSOR_MODEL" . Is it wrong ?

3. Comments on users's manual version 0.9b

The User manual version 0.9b was submitted to the argo-dm-format@ifremer.fr mailing list on the 5th of march 2002.

From : Roger Goldsmith

Date : 06/02/2002

Thierry Carval,
some edit for the manual.

1) There still doesn't seem to be a consensus on whether it's
POSITIONING or POSITIONNING
It is used both ways throughout the document, sometimes one way
in the variable and another in the attribute.
TC: we now use the right spelling is POSITIONNING.

2) p. 32 PTT: ORBCOMM should maybe have two MM throughout?
TC : ok

3) p. 33 LAUNCH_QC has attributes LAUNCH_DATE_QC
TC : ok

4) p. 33 START_DATE_QC has a long name of launch date.
TC : ok

5) p. 27 JULD_START_TRANSMISSION
the attributes have JULD_DESCENT_END
the example has DATE_TIME format rather than the double type.

TC : ok

From : Yasushi Takatsuki

Date : 22/02/2002

Dear Thierry and All,

[1] I found that 'NOMINAL_PROFILE_PRES', 'NOMINAL_PARKING_PRES' and 'PARKING_PRES_QC' field were deleted in format version 2.0, that were existed in version 1.2b.
I think these information are valuable especially for the floats that may change parking/profiling pressure on each cycle.
Therefore, I propose to adding following definitions in the Cycle data block;

```
int NOMINAL_PROFILE_PRES(N_CYCLE);
    NOMINAL_PROFILE_PRES:Comment ="Nominal pressure of the ascending
profile for this cycle";
    NOMINAL_PROFILE_PRES:_FillValue = 99999.f ;
    NOMINAL_PROFILE_PRES:units = "decibar";
int NOMINAL_PARKING_PRES(N_CYCLE);
    NOMINAL_PARKING_PRES:Comment ="Pressure of the submarged drift for
this cycle";
    NOMINAL_PARKING_PRES:_FillValue = 99999.f ;
    NOMINAL_PARKING_PRES:units = "decibar";
char NOMINAL_PARKING_PRES_QC(N_CYCLE);
    NOMINAL_PARKING_PRES_QC:Comment = "Quality on parking pressure ofthe
cycle";
    NOMINAL_PARKING_PRES_QC:Convensions="Q where Q=[0-9]";
    NOMINAL_PARKING_PRES_QC:_FillValue = "0";
```

TC : nominal values are set before launching. They are located in the float cycle informations section of the meta-data format. I think that there is no need to use the word NOMINAL in this section (these parameter are here naturally nominals).

[2] In the Profile format, we define "PSAL_CORRECTED_ERROR" to store estimated error for the salinity correction, but no definition in the trajectory format. Is it unnessesary for the trajectory format?

TC : psal corrected values will probably not appear in a near future in the trajectory format. That is the reason why it is not mentioned. When corrected values will be available, they will be added.

[3] Should we treat the measurement field (PRES, TEMP, PSAL, CNDC) for the floats that do not measure during the surface/subsurface drift?
(Many APEXs measure only profiles and have no drift data.)

1) Remain all the definition as same in profile format for the float and filled with _FillValue except for PRES? (PRES may be set to 0.)
2) Only defined PRES/PRES_CORRECTED... and filled with 0 for PRES?
3) Do not define PRES/TEMP/PSAL/CNDC and N_PARAM,TRAJECTORY_PARAMETERS for that float?

TC: I am in favour of the third option which is probably the most pragmatic.

What do you think about above things?

From : Mark Ignaszewski

Date : 22/02/2002

Question regarding the Trajectory data file format: Should the DATA_TYPE be set to something similar to "Argo trajectory data". (In the example file it is "Argo float".)

This would seem to be more consistent with the other types where:
      profile file: DATA_TYPE = "ARGO PROFILE"
      technical file: DATA_TYPE = "ARGO TECHNICAL DATA"
TC : ok

From : Mark Ignaszewski

Date : 26/02/2002

Part of the GDac design is to validate the format of the data files received from the Dac. I have to show my ignorance of netCDF and ask what it means to be "correctly formatted". Specifically:

1) Does every element defined in your format description HAVE TO BE in the data file, even if there is no data for it currently? For example, if there is no "CNDC_CORRECTED" profile for a float, does it need to be defined in a Dac data file? (Should the GDac reject the file and return it to the Dac if it is not there?)
TC : the number and the name of the parameter is set in N_PARAM and PARAMETER. If a parameter exists, it has an associated corrected value. If there is no correction available, the corrected value is equal to the value.

2) (I think it is obvious but I will ask) The only "dimensions" and "variables" in a Dac data file should be those defined in the format description. Is that correct? (Should a Dac data file containing a variable not in the format definition be rejected?)
TC : if a variable of a file is not in the format, then we consider that the file format is not correct.

As I was working on some software to do the validation, I uncovered two things:

1) The Profile version 2.0 definition on your Web page contains the variable "Q_PROFILE_PSAL_QC". I think it should be "PROFILE_PSAL_QC"
TC : corrected

2) The example data file you sent me contains the variable "POSITIONNING SYSTEM" (notice the NN). I think it should be "POSITIONING_SYSTEM".
TC : ok, corrected

From : Bob Keeley

Date : 04/03/2002

Dear Thierry and Yasushi,
   I am just back from a meeting, so have not had time to think about this.
I was hoping to see a reply from Thierry, but lacking that I thought I would give my first thoughts about this anyway. I am concerned about modifying yet again the format that we agreed to in Brest. We simply cannot be continually doing this unless it is absolutely necessary. So, unless we

have any floats either in the water or expected in the water in the next year that will modify their parking pressures from one cycle to the next, I would want to delay this.

If this is not the case, we have another place, perhaps, where this information could go. We do have the technical file in which we record all of the extra things for each cycle, such as battery voltages or whatever else we have. Considering that knowing the nominal parking pressures will not really influence how the data are viewed by a user, we could simply put this information in the technical file.

Regards, Bob

From : Mark Ignaszewski

Date : 06/03/2002

I noticed a few more things as I was working on the netCDF files.

1) In your "profile v2.0 specfication" on the Web page all of the HISTORY_xxx variables are dimensioned as

```
      (N_PROF, N_HISTORY, ...)
```

TC : corrected

They must be (N_HISTORY, N_PROF, ...)
Your example file had this correct of course.

2) In the example file the definition of HISTORY_QCTEST is:

```
      float HISTORY_QCTEST(N_HISTORY, N_PROF, STRING16) ;
```

This should be "char" instead of "float", I believe.
TC : ok

From : Yasushi Takatsuki

Date : 08/03/2002

I'm sorry to delay my response.

Page 9,
Ucar web site address should be "http://www.ucar.edu/ucar/".
TC : ok

Page 10, Comment of N_PARAM
N_PARAM should takes a value of 2/3/4 instead of 1/2/3 as in other format.
TC : ok

Page 11, Comment of PI_NAME
PI means 'principal investigator", not "principle".
TC : ok

Page 22, Comment of REFERENCE_DATE_TIME
"January" instead of "january".
TC : ok

Page 24, Comment of LONGITUDE
It should be removed '-' from '-16.7222' for 16-43.1992 E.
TC : ok

Page 26-27, Comment of JULD_xxxx_xxxx
Examples should be 18833.98733.. or something else instead of "20011230090500 : December ...".
TC : ok

Page 43, Physical parameter codes (GF3)
Concerning to the "Units", it should not be inconsistent in the same document.
If we use "mmho/cm" for conductivity,
it should be removed "Units" information from Table 4.3.
But we strictly follow the GF3 manner, we should use "mhos/m" instead of "mmho/cm" in the Argo data.
The unit used in the Argo data is not depend on the unit used in Database at each institution.
TC : ok, we shall follow GF3 and use "mhos/m"

Page 49, http address of US-GODAE server
Does anyone know correct address ?
TC : ok

From : Mark Ignaszewski

Date : 08/03/2002

> Page 49, http address of US-GODAE server
> Does anyone know correct address ?

The address is: www.usgodae.fnmoc.navy.mil
TC : ok

Do not be alarmed that there is no mention of Argo on this Web site at this
time. Links to the US Argo GDac will be on the site very soon.

From : Roger Goldsmith

Date : 12/03/2002

Re: DIRECTION: Convention ...

A descending float may look strange but the DIRECTION would seem to be Needed to indicate the direction a particular profile was taken, even though everything may be stored in conventional top to bottom. Especially true if you are keeping profiles in separate files.
The ORBCOMM floats take a profile on the way down the first time, Profile on the way up, transmit both profiles then continue with only ascending profiles for the rest of the mission. Is this a BOTH? That seems like a good way to mess up counters. If it is ascending one has to account for the first profile somehow and it's not a separate CTD cast.
TC : the direction of each individual profile is indicated in the profile format. The DIRECTION in the meta-data format is maybe useless. Maybe can it be removed in the next version of the format.

From : Stephen G. Loch

Date : 13/03/2002

My comments relate mostly to typos or phrasing but there are one or two substantial issues too.

Re metada format Roger's proposal (using the output from ncdump) sounds the simplest but to assure myself that it was OK I would need to see whether the visual appearance (of STRING256) was satisfactory or accept somebody else's word for it. Are we allowed new-line characters?

Steve

S.G.Loch
BODC Systems Coordinator
www.bodc.ac.uk

Timing Issues

UTC is Coordinated Universal Time (or if you like Universal Time Coordinated - note the 'd' on the end, pace Claudia). The French name, which I was tempted to think gave rise to the inversion of the natural English, is Temps Universel Coordonne (plus acute accent). So if explaining what UTC is use 'Coordinated Universal Time'.

The example given for relative Julian day number on p13, unlike p26,27, is correct but needs to go further (note: eliminate the blanks preceding the colons):
Suggest putting: 18833.8013889885 = July 25 2001 19:14:00
TC : ok

Are we required to use 1950.1.1 or can we do what we like? In the latter case you just extend the example (giving an arbitrary origin) but with the former a comment in the preamble would be useful. The obvious date to use for newly deployed floats is 2000.1.1.
TC : in version 2.0, we shall stick to 1950.1.1 . If we want to change it, we shall wait a version 2.1.

Most of the comments re-inforce the idea of using UTC. Some of the comments on DATE_TIME parameters don't though. It is important in a global setup to emphasise that UTC applies as much to the metadata as to the data.
TC : ok

Issues of Style

Arguably you should capitalise 'celsius' and 'julian' but the document seems consistent here so leave it.

I guess some measure of consistency is useful on American/British spelling as you can have both on the same page or even line at the moment. Standardising on 'centre' would dispose of many of these (and underline the global nature of the enterprise) and agrees with the names of defined variables.
TC : "center" has been replaced by "centre" troughout the document.

String Dimensions

Sorry, but I don't know what a null terminated (null-terminated) string dimension is. If the string length is appropriate to say STRING4, then you can store 4 characters in an instance of the NetCDF variable so why do we need to say 'null-terminated'? In C you actually need 5 characters of memory in a program because of the null (binary zero byte) that terminates a string but that has nothing to do with the NetCDF file and is not a requirement for, e.g., Fortran. If we are saying that all trailing blanks should be converted to zero then say so with a particular note to Fortran programmers. Of course this doesn't save any space in the file. Or is it saying, as I suspect is the case, that we must reserve at least one character to allow a binary zero to be inserted at the end of the line, to make programming simpler in C (having retrieved the array no further shifting is required to produce all the strings)?

If the last interpretation is the correct one then DATA_CENTRE(STRING2) (p40) and perhaps other variables are underdimensioned. E.g. HISTORY_PARAMETER. The quickest way to solve the problem, if there is one, would be to add 1 to the string dimensions but keep the same names!

TC : "null terminated" has been removed from comments.

Versions

========

Why do have some VERSION variables as STRING4 but one or two as float? Surely better STRING4. E.g p9.

TC : yes, but we shall wait for a nex version of the format.

Typos

=====

'The GF3' crops up many times as does 'the The GF3': replace with 'the GF3'. There are important modifications to be made on p36 which until they are made prevent preclude authoritative publication.

Versions

========

Why do have some VERSION variables as STRING4 but one or two as float? Surely better STRING4. E.g p9.

TC : ok.

Page by page

============

p1 'Stephan Loch' --> 'Stephen Loch'

TC : ok.

p4 'Positionning' --> 'Positioning'

TC : ok.

p6 'data-format --> 'data format'

TC : ok.

p7 'An argo profiler cycle' --> 'An Argo profiler cycle'

TC : ok.

p7 'argos transmission' --> 'Argos transmission'

TC : ok.

p9 'NetCDF' appears as :'NetCDF', 'netCDF' and 'NetCdf'. Suggest 'NetCDF'.

TC : ok.

p9 'General Informations' should be "General Information'

TC : ok.

p9 'ascii' is usually 'ASCII'

TC : ok.

p11 'acceptable data type' --> 'acceptable data types'
TC : ok.

p11 As noted above HANDBOOK_VERSION is given as float; suggest STRING4
TC : ok but in a next version.

p11 'january' --> 'January'
TC : ok.

p11 'temperature in degree' --> 'temperature in degrees'
TC : ok.

p12 'measurement occur' --> 'measurement occurs'
TC : ok.

p12 'for data center' --> 'for data centre'
TC : ok.

p12 Surely ':comment' should be ':long_name'
TC : ok but in a next version.

p12 'different identifiers schemes' --> 'different identifier schemes'
TC : ok

p12 box horizontally truncates last line of text at bottom of page on my printout
TC : I did not notice. Maybe is it a printer problem.

p13 'location of profile is': the 's' seems to be printed on top of the 'i'
TC : I did not notice. Maybe is it a printer problem.

p13 JULD and JULD_LOCATION examples need improving (see above)
TC : ok

p14 Suggest 'Each item of' be replaced by 'Each NetCDF variable in'
TC : ok

p15*2 'the The GF3' --> 'the GF3' (as noted above)
TC : ok

p15 <PARAM>_CORRECTED attributes are given for salinity. The comment therefore needs to say: 'Attributes given here are for salinity and should be changed appropriately for other variables'.
TC : ok

p16 another 'The GF3'
TC : ok

p16 'to a parameter of a profile' --> 'to a parameter in a profile'
TC : ok

p16: No examples of equations given. Surely we can manage a linear calibration?!
TC : ok

p17 data center' --> 'data centre'
TC : ok

p17 'An history record ' --> 'A history record'
TC : ok

p17 'refernce' --> 'reference'
TC : ok

p18 There looks to be unfinished business here (remarks in italics)
TC : ok

p20 '17 :28 :34' --> '17:28:34' (as noted above)
TC : ok

p22 'january' --> 'January' (as noted by Yasushi)
TC : ok

p22 Should ARGO be capitalised. It's not an acronym, is it? Suggest Argo.
TC : ok

p22: GF3 again
TC : ok

p23: Horizontal truncation of text at foot of page
TC : I did not notice. Maybe is it a printer problem.

p24 colon spaces
TC : ok

p24 'different identifiers schemes' --> 'different identifier schemes'
TC : ok

p24 'refrence' --> 'reference'
TC : ok

p24 'Argos locations classes' --> 'Argos location classes'
TC : ok

p25 <PARAM>_CORRECTED etc. needs comment on required adjustment for other variables (see p15 comment)
TC : ok

p25 Example is inappropriate. Need to add: 18991.37872222 = December 30'th ....
TC : ok

p27*4 Ditto
TC : ok

p28 'Date' --> 'Date and time (UTC) of action
TC : ok

p28 HISTORY_PREVIOUS_VALUE. Are all flags numeric and are they going to stay that way? If not the type of this variable is a problem. Secondly is it unambiguous whether you are referring to the flag or the parameter? Cannot they both change at the same time? What about GF3 code 5?
TC : we may decide to change in the next version of the format.

p29 It might be helpful to say 'quality control' rather than the shorthand 'qc'
TC : ok

p31 Type of HANDBOOK_VERSION (suggest char STRING4).
TC : ok but in version 2.1

p30 Remove spaces before time colons, 'ascii' --> 'ASCII', UTC expansion
TC : ok

p31 Ditto
TC : ok

p32 'chracteristics' --> 'characteristics'
TC : ok

p32 'multi-beacons' --> multi-beacon'
TC : ok

p32*2 'is equiped' --> 'is equipped'
TC : ok

p33 'ARGO' --> 'Argo'
TC : ok

p33 'data center' --> 'data centre'
TC : ok

p33 ANOMALY. I think this is an unfortunate name as it has widespread use in physical oceanography. What's wrong with COMMENT?
TC : we may decide to change in the next version of the format.

p33 'the immesion drift' --> 'the immersion drift'
TC : ok

p34 time colon spaces
TC : ok

p34 'deployement platform' --> 'deployment platform'
TC : ok

p34 'ctd or xbt' --> 'CTD or XBT'
TC : ok

p35*2 'The GF3' --> 'the GF3'
TC : ok

p36*n 'PREDEPLOYEMENT' --> 'PREDEPLOYMENT'
TC : ok

p36 No example equations, coefficients or comments
TC : ok

p37 'Time spent to ascend' --> 'Time spent in ascent'
TC : ok

p37 'Time spent to descend' --> 'Time spent in descent'
TC : ok

p38*3 Units in this document are singular in the comments field. Suggest 'decimal hours' --> 'decimal hour'
TC : ok

p38 Text horizontally truncated by box at bottom
TC : I did not notice. Maybe is it a printer problem.

p39 'Thes information are registred' --> 'This information is registered'

TC : ok

p39 'acii' -->'ASCII'

TC : ok

p39 'universal time coordinate' --> 'Coordinated Universal Time', 'ascii' -->'ASCII'

TC : ok

p39 'january' -->'January'

TC : ok

p40 See comment about underdimensioning given above for DATA_CENTRE and DATA_TYPE. 'Argo technical data' is 19 characters and so STRING16 appears to be too small anyway.

TC : ok but in version 2.1

p42. On the subject of ncdump, I have recently found a bug in ncdump which can give rise to faulty listings when the C_format attribute is present. How the bug has lasted so long defeats me but perhaps because people don't report these problems. Anyway if Boulder don't fix it the patch is small and can be made available.

p43 As Yasushi has pointed out GF3 uses mho/m (surely this rather than mhos/m) and not mmho/cm.

TC : ok

p43 Should RCRD be in 4.3?

TC : sorry, I do not understand the question

p44 Shouldn't BODC feature as a data centre? How come we have RU and VL? They should be mutually exclusive so the wording for RU needs adjusting. OK, so it's a MEDS problem!

TC : ok

p45 Additional interposed blank line appearing (for 1,2 & 3). Should be removed.

TC : ok

p46 It would be helpful if DS was noted as being the concatenation of level and class.

TC : ok

p46,47 The notes are referenced by number but they are not numbered. Just add 1), 2), etc. to the start of the relevant paragraphs.

TC : ok

p47 'geospacial' --> 'geospatial' (but some dictionaries allow it)

TC : ok

p48 'occur.at' -->'occur at'. 'profile (' --> 'profile. ('

TC : ok

p48 '4.9 Positionning' --> '4.9 Positioning'

TC : ok

p48 'centers' --> 'centres'

TC : ok

p49 'delayed' --> 'delayed mode'

TC : ok

If people are unsure what the issues are relating to string dimensions, variables (and attributes) , they can look for a fuller account in section 7.15 of the NetCDF User's Guide for C, Version 3, June 1997.

Incidentally Russ Rew of Unidata accepts there is a bug in ncdump (and a couple of other places as well he thinks) and hopes to fix the problem for the next NetCDF beta release (3.5.1).

From : Mark Ignaszewski

Date : 15/03/2002

In the Profile File format:

1) The JULD:units and JULD_LOCATION:units both say
"days since 1995-01-01 00:00:00 UTC"

They should probably say
"days since REFERENCE_DATE_TIME"

TC : we shall discuss it for version 2.1

2) In the Definition of CALIBRATION_DATE, it should be
char CALIBRATION_DATE (N_PROF, etc...

TC : ok

From : Bob Keeley

Date : 25/03/2002

In the latest version of the User's Manual, certain information seemed to have been dropped. I am writing a detailed description of what changes in file contents we expect to see as a result of delayed mode QC. In the course of this, I have been checking the User's Manual and see that there is no table of Action Codes. I am sure this must have been there before.

TC : they have been transferred in the chapter 4.7 of the refernce tables.

I also noted that in the description of fields, where code tables entries are required, though the comments refer to the tables, they do not refer to the exact table. I would suggest that the exact table to look at be linked to the appropriate fields. For example, the comments for HISTORY_INSTITUTION should refer to table 4.4.

TC : ok

Another note, we will be having our scientists in two different places in Canada carry out the delayed mode QC. Consequently, we will need two more entries in table 4.4 I propose to use CI for Institute of Ocean Sciences, Canada, and to use CB for Bedford Institute of Oceanography, Canada. Can you please add these entries to your list?

TC : ok

I also note that the list of GF3 codes, table 4.3, does not contain codes needed to describe dates and times in history records when these are required. Specifically we require LAT$, LON$ and DAT$.

TC : is there a GF3 code for these ?

I will send a copy of what I write since I think this will be a handy thing for all of the national centres to use. We can generalize the contents for international consumption.

From : Naveenta Anand

Date : 02/04/2002

I have the three formats ready as per the differences indicated in your last email. Please run them through your program one more time. You will have to fix one thing in your program as per follows.

The DATA_TYPE in technical format has a dimension of STRING16 according to the manual. It should be STRING32 as we would like to display "Argo Technical data" -which is > 16 characters. Thus String 16 is too short for displaying the same. My program assumes a dimension of STRING32 currently. You might want to change your verification program accordingly.
TC : ok for 32 in technical data section.

From : Mark Ignaszewski

Date : 04/04/2002

Regarding the size of DATA_TYPE: The Argo Data Management community must address the issue of "flexibility" in the file formats more specifically.
I
certainly don't want to put myself in the position of setting policy regarding file formats.

The published format specification defines both the size of the DATA_TYPE element and its allowed settings. It may be reasonable to allow variables dimensioned with the STRINGxx dimensions to be longer than the specification but I would like to hear the opinions of others involved in the Data Management group.

Opinions?
TC : ok, bt we shall wait version 2.1

(I too have a couple of variables I am tempted to change unilaterally but I have not yet. Specifically, I think HANDBOOK_VERSION should be "char (STRING4)" not "float" to be consistent with FORMAT_VERSION.)
TC : ok, but we shall wait version 2.1

From : Mark Ignaszewski

Date : 05/04/2002

I have a few comments regarding the formats (I apologize to Thierry if he has already seen these before):

1) The "units" attribute for all of the "JULD" related variables should be
      "days since REFERENCE_DATE_TIME"
not
      "days since 1950-01-01 00:00:00 UTC"
TC : ok, bt we shall wait version 2.1

2) The "_FillValue" value for the "JULD" related variables should be consistent in all of the formats. It is "999999." in some and "99999." in others.
TC : the adopted fill value is "999999."

3) The "units" attribute for the TEMP variable needs to be consistent. It is defined as "degree_Celsius" in some cases and "degree_celsius" in others

TC : the unit name is "degree_Celsius"

4) The HANDBOOK_VERSION variable should probably be defined as

```
      char HANDBOOK_VERSION (STRING4)
not
      float HANDBOOK_VERSION
```

TC : ok, but we shall wait version 2.1

From : Denis Croize-Fillon

Date : 12/04/2002

* 'Conventions' ou 'conventions' car dans ce format, les deux ecritures sont presentes. Bien sûr, quand il fallait 'C' j'ai trouvé 'c' dans le fichier et inversement. En ce sens, le doc ne me parait pas cohérent, le format Argo Profile File Format 2.0 ne contient que 'conventions',
* il y a un certains nombre de variables sans dimension (1 par defaut mais ...) : TRANS_REPETITION, CLOCCK_DRIFT, DIRECTION, ... Pour en avoir fait l'experience dans le developpement de la lib sous matlab, on s'attend a avoir des variables parfaitement definies et non pas définies par defaut. Il me faut me resoudre a traiter ce cas mais il me semble que, ainsi définies, ces variables sont incohérentes,
* un détail, CLOCK_DRIFT etant du type float, _FillValue et de type float et non pas char (les guillemets sont en trop).
TC : we use “conventions” nstead of “Conventions”

From : Claudia Schmid

Date : 22/04/2002

Yeun-Ho found an inconsistency in the format descriptions in the user manual 01/03/02 (version 0.9b) p.26. (trajectory format):

JULD_ASCENT_START has convention YYYYMMDDHHMISS and is defined as double. I think you wanted to put the same text here than p.24 for JULD.

Same for: JULD_ASCENT_END JULD_DESCENT_START JULD_DESCENT_END JULD_START_TRANSMISSION
TC : JULD_xxx are declared as double. The comment “Format : YYYYMMDDHHMISS” is now deleted.

From : Argo Science team

Date : 23/03/2002

These comments were reported by Sylvie Pouliquen for the Argo Science Team (Hobbart meeting, march 2002).

Page 16 : update the explanations on profile calibration. Only the best scientific calibration is stored.
TC : in version 2.0, different calibrations are stored for each profile. In version 2.1, only the best calibration will remain. The N_CALIB dimension will be removed.

4. Comments on user’s manual version 0.9c

The User manual version 0.9c was submitted to the argo-dm-format@ifremer.fr mailing list on the 24th of april 2002.

From : Mathieu Belbéoch

Date : 24/04/2002

Sorry to bother you again with the float Ids but I recall that this unique
WMO Id is A9nnnnn and NOT QA9nnnnn.
When BUFR wil be implemented the Q will disappear ...
If it's impossible to remove the Q in your internal processing system,
please think to use the standard 7 digits WMO Id on your websites.
Thanks for your comprehension,

TC : in march 2002, during IAST meeting in Hobbart it was decided to use QA9IIIIII as the standard argo float ID.

From : Claudia Schmid

Date : 24/04/2002

I agree with Matthieu concerning the treatment of Q as part of the WMO ID.
I think we should not do that. I think it's better to change it now then
to wait until we start using BUFR.
TC : see above

conductivity: I'm still not sure why we use mhos/m which is not an SI unit.
TC : mhos/m is the unit of GF3 CNDC code.

Here are some editorial comments (I only looked more closely at the red text, and at UTC):

p. 9: ... all dates and times have to be given in UTC (universal
time coordinates)
p. 13: universal time coordinates (not coordinate) (occurs twice)
p. 16: universal time coordinates (not coordinated)
p. 20: universal time coordinates (not coordinated)
p. 24: universal time coordinates (not coordinate)
p. 30: universal time coordinates (not coordinated)
p. 39: universal time coordinates (not coordinated universal time)
TC : corrected

From : Thierry Carval

Date : 03/07/2002

We agreed on the content of argo meta-data files with argo user's manual version 0.9c .
We now have to clarify the format of these meta-data files.

1. A Netcdf format as defined in the user manual
2. An ascii format that is a ncdump of the netcdf format
3. A fixed length ascii format not defined yet

I personnaly prefer the option n.2 : a ncdump version of the netcdf file.

I join a sample generated by JMA.

TC : there is no clear opinion about what to do. JMA generates NetCDF meta-data files, MEDS generates ascii 'ncdumped files'. Let's consider that option 1 and 2 are valid until the next argo data management meeting in September 2002 (Ottawa).

5. Comments on user's manual version 1.0

From : Roger Goldsmith

Date : 12/03/2002

Re: DIRECTION: Convention ...

A descending float may look strange but the DIRECTION would seem to be Needed to indicate the direction a particular profile was taken, even though everything may be stored in conventional top to bottom. Especially true if you are keeping profiles in separate files.

The ORBCOMM floats take a profile on the way down the first time, Profile on the way up, transmit both profiles then continue with only ascending profiles for the rest of the mission. Is this a BOTH? That seems like a good way to mess up counters. If it is ascending one has to account for the first profile somehow and it's not a separate CTD cast.

TC : the direction of each individual profile is indicated in the profile format. The DIRECTION in the meta-data format is maybe useless. Maybe can it be removed in the next version of the format.

From : Stephen G. Loch

Date : 13/03/2002

Are we required to use 1950.1.1 or can we do what we like? In the latter case you just extend the example (giving an arbitrary origin) but with the former a comment in the preamble would be useful. The obvious date to use for newly deployed floats is 2000.1.1.

TC : in version 2.0, we shall stick to 1950.1.1 . If we want to change it, we shall wait a version 2.1.

Versions

Why do have some VERSION variables as STRING4 but one or two as float? Surely better STRING4. E.g p9.

TC : yes, but we shall wait for a nex version of the format.

p12 Surely ':comment' should be ':long_name'

TC : ok but in a next version.

p28 HISTORY_PREVIOUS_VALUE. Are all flags numeric and are they going to stay that way? If not the type of this variable is a problem. Secondly is it unambiguous whether you are referring to the flag or the parameter? Cannot they both change at the same time? What about GF3 code 5?

TC : we may decide to change in the next version of the format.

p33 ANOMALY. I think this is an unfortunate name as it has widespread use in physical oceanography. What's wrong with COMMENT?

TC : we may decide to change in the next version of the format.

From : Mark Ignaszewski

Date : march 2002

The published format specification defines both the size of the DATA_TYPE element and its allowed settings. It may be reasonable to allow variables dimensioned with the STRINGxx dimensions to be longer than the specification but I would like to hear the opinions of others involved in the Data Management group.
Opinions?
TC : ok, bt we shall wait version 2.1

From : Thierry Carval

Date : march 2002

The trajectory format is defined for one platform.
To disseminate argo daily trajectories (and for other purposes), we would like to use a multi-platform trajectory format.

From : Argo Science team

Date : march 2002

These comments were reported by Sylvie Pouliquen for the Argo Science Team (Hobbart meeting, march 2002).

Page 16 : update the explanations on profile calibration. Only the best scientific calibration is stored.
TC : in version 2.0, different calibrations are stored for each profile. In version 2.1, only the best calibration will remain. The N_CALIB dimension will be removed.

From : Yeun-Ho Chong

Date : 13/08/2002

For writing a "meta" netcdf file, what are the fill values for 'sensor_accuracy', 'sensor_resolution', and 'trans_frequency' when we do not have those info?

TC :
SENSOR_ACCURACY:_FillValue = 99999.f;
SENSOR_RESOLUTION:_FillValue = 99999.f;
TRANS_ FREQUENCY:_FillValue = "99999";
.

From : Bob Keeley

Date : 16/08/2002

In working at using the new trajectory netCDF format for the profiling float data for the WOCE DVDs, I found a couple of small problems. They relate to stated valid_max and Valid_min. You should check this in the profiles files as well.
1. temp.valid_max must be greater than 40. We have seen data in the Red Sea
that was 41+ degrees. I suggest this upper limit change to 45.
2. Temp_correct.valid_max - same comment as for temp.

TC : OK

.

From : Thierry Carval

Date : 03/09/2002

```
Trajectory format :
PRES, TEMP parameters should not be explicitly mentioned in the format.
The <PARAM> , <PARAM_CORRECTED> and <PARAM_CORRECTED_QC> specification is
enough to specify these parameters.
The previous remark also applies to profile format specification.
```

From : Denis Croize-Fillon and Thierry Carval

Date : 03/09/2002

```
Measurements in trajectory format : most of the argo profilers do not
perform measurements during surface drift.

In that case : N_PARAM = 0
TRAJECTORY_PARAMETER entry is not created
<PARAM> entries are not created


The N_MEASUREMENT dimension is not the “number of recorded locations and
measurements of the file” but the “number of locations” (the name of this
dimension is misleading). I suggest to rename it “N_LOCATION”
```

From : Charles Sun

Date : 03/09/2002

```
First of all, I would like to congratulate you for making the version
2.0 of Argo data formats available.

I have very carefully reviewed the Version 2.0 formats and have some
Concerns about the formats. To make my concerns short, a NetCDF file
should be reversible, namely, convertible by using the Unix/Linux "ncdump"
and "ncgen" programs.
Argo V2 format files do not meet this criterion, at least in my view.
Other concerns include naming conventions which may cause some confusions.
I'll present my findings to the group at the coming Argo DM meeting at
Ottawa, if Bob can allocate about an hour for me. Otherwise, I can discuss
it with you off-line.

As a quick starting point for fixing the Argo V2 format problem, I
downloaded one of real-time Argo data from Japan DAC (I assumed the format
is the same as the one used at IFREMER) and modified the format for a
better display, at least, by using ncBrowse. Two nc files are attached
here. The "RQ5900229_003.nc" was downloaded from Japan DAC and the
```

"RQ5900229_003r.nc" is the revised version.
If you can not open them in your mail client program, you could ftp them from NODC's ftp site at ftp.nodc.noaa.gov. The files are located at the "pub/outgoing/argo" directory or you could read the files over the Internet by using ncBrowse and specify the Web URL as http://sunspot.nodc.noaa.gov/argo/data and Data File URL as either http://sunspot.nodc.noaa.gov/argo/data/RQ5900229_003r.nc or RQ5900229_003.nc

.

From : Ann Thresher

Date : 09/09/2002

In meta-data format : “units value : decimal hour <> decimal hours suggest you change the manual on this one - 8 decimal hour makes no sense whereas 8 decimal hours does.... But I can change mine if you insist... "

From : Denis Croize-Fillon

Date : 10/09/2002

.

All variables should be defined with a dimension even if the dimension is 1.
Matlab requires a dimension to create a variable.
Exemple :

```
        DIM1 = 1;
        float HANDBOOK_VERSION(DIM1) ;
                HANDBOOK_VERSION:comment = "Data handbook version" ;
```

From : Rebecca McCreadie

Date : 30/09/2002

I've just been
going through the Argo User Manual and have found a reference to a table that doesn't exist. On page 17 references Table 4.7. I can't find this table.
TC : the correct reference is Table 3.7

From : Ann Thresher

Date : 21/10/2002

I am just about to generate what I hope will be (nearly) the final version of our netcdf files. One issue remains truly outstanding - how have we decided to handle the QC arrays? I was told at the meeting that concatenated strings were wrong and have since been told that adding a string dimension (the only way to get arrays) is also wrong and we were sticking with long strings because that was how the dimensions were specified in the handbook. So - we all need to know which way the wind has blown. I can easily modify my code to do either.
Thanks for all the help! Once I get this code "fixed" I can start generating automatic feeds to the GDAC's and cross this off my list!!

From : Bob Keeley

Date : 22/10/02

I agree with Ann and others that having the QC flags as a string rather than an array is at least inconvenient. If it were not that there are more thana few folks who have written software to conform to the current form, I would be pushing hard to make the fix. It seems to me that there are a number of things we should weigh before we decide.

1. Is it only an inconveniance or really a problem having QC flags as strings?
2. How long will it take to get the corrected format settled?
3. How willing are those who software is now working to make changes?
4. What will be the impact on users of the GDACs now if we change?
5. If we are intending to make other alterations in the format can they be done now or later? When is the best "later" time to select?
6. Other issues?

I have a few opinions on this, but I would like others to voice there's. So, here are mine.

1. I am soft on this. At present I view it as an inconveniance more than a problem.
2. I would guess it would not take long.
3. We are willing to make changes.
4. If there are only a few, the impact is light. We need GDAC comment.
5. The only other issues in the format are, I think, relatively minor. If we could accommodate them and do it all now, I think now is a better time before use becomes heavy.

From : Charles Sun

Date : 22/10/2002

I think the Argo Data Format Group should ask other experts who are familiar with the netCDF file structure to test the current version (V2.0) and take their feedback for possible revision.

The QC flags as a string or an array does not matter, but should be Consistent with the manual or we have to revise the manual to be consistent with the data.

NODC has developed a procedure to download and convert the Argo profile NetCDF file (V2) to the plain text, comma separated format Just let us know when the current format is changed.

TC :
Here are 3 answers to your 3 remarks.

1. Argo data format should be reviewed by other experts :
Since 2000, the Argo formats have been reviewed by people from argo data management group, but also by people (some of them experts) from other communities such as PMEL, Unicar, LAS.
Any feedback is interesting. For the moment, I did not receive any comments on a major problem that implies an urgent revision.

2. QC flags are not consistent with the manual :
I think that I do not understand your remark. QC flags from AOML, JMA, MEDS and Coriolis are consistents with the manual.

3. Let us know when the current format is changed :

A revision of the format will not be enforced before the next Argo data management meeting in september 2003 in Monterey.
So, I think that you can confidently start to use the NODC procedure to download and convert the Argo profile netCDF file (V2) to the plain text.

From : Garry Dawson

Date : 28/10/02

I have two suggested amendments for the User's Manual Ver 1.0 (you may already be aware of these)
1. The reference tables are all numbered starting with 3. but in section 2 the tables are described as 4.2 etc . See 2.2.4 <PARAM> for an example.
2. The table at 3.7 refers to XBT in several places - I assume this is because it is a copy of the original table of GTSPP History Codes Table but in an ARGO Manual it could be confusing without explanation.
TC : ok

From : Bob Keeley

Date : 09/12/2002

Here are my views regarding the CNDC issues. First, I view the inclusion of valid_mins and maxs as optional. They depend on the waters sampled and the data reported in the file. I can see two different profile files having two different mins and maxs simply because they refer to the individual file contents. I do not view these as QC ranges to validate the observed values.

To be honest, I would not have mins and maxs listed for observed variables. I note that TEMP has a valid max of 40, but that is clearly wrong in the Red Sea at times. Likewise a valid min of -3 clearly does not cut off impossible low temperatures. I would just as soon remove these from all <param> fields.

The units is a different issue. We want to specify units in which the data are reported to remove conversion problems later. If the number of decimals is insufficient to report to the units specified, this is a problem in the format specifications and should be altered.

From : Bob Keeley

Date : 09/12/2002

I have the following to say regarding Charles comments and Mark's reply.
1. First, I understand it that Charles is talking about the multi-profile files and not the original files sent to the GDACs. The important issue here is to identify where the problems noted by Charles are created. Having said this, I would like to address the issue of history and calibration groups. In Canadian floats, there is no calibration records so N_CALIB should be set to 0. This is our mistake and will be fixed. If N_CALIB=0 I don't see why we should be filling every field. I don't image this is what Charles means. I assume he means that if a group is present, every field of the group should have useful information or missing values.

I agree with this.
Regarding history groups, I think we should be seeing meaningful records here. If this is not the case in the original files sent to the GDAC this is a problem we must correct. Responding to Mark's item 1 I would want to see all of the history fields being mandatory.
2&3. Mark raises issues that have to do with the creation of the multi-profile files. My understanding of netCDF is that if you have two separate profiles with say two different sets of depths (one has 53 levels and one has 55 levels, say), the combined file must have the union of the set of levels, that is at least 55 levels (and perhaps more if there is a level in the first profile not present in the second and vice versa). I think this applies everywhere. This means the multi-profile files can get quite large, quite fast. If we don't do this, I believe that the netCDF software that is available will have troubles with the files and need special handling. Please correct me if I am wrong.

From : Thierry Carval

Date : 09/12/2002

As far as I understand, the data format issue about valid min, valid max and resolution is this one :
can we find unique values for valid min, valid max and resolution that are correct for all dacs, all oceans, all floats, that will not change in time ?
If the answer is yes, then these 3 values will be recorded as the part of the data format.

If there are not such unique values, then they will not be part of the data format.
Each dac will decide what values of valid min, valid max and resolution he will use for CNDC, TEMP and other parameters.

From : Takashi Yoshida

Date : 10/12/2002

In Japanese floats, regarding the issue 1), missing information problems are created by JMA because I haven't understood well how those fields should be filled properly.

DC_REFERENCE, PARAMETERS, SCIENTIFIC_*, CALIBRATION_DATE and HISTORY_QCTEST values are missing in the Japanese profile data. DC_REFERENCE and HISTORY_QCTEST can be filled with approprate values. However, it seems to be difficult to fill the values which have N_CALIB dimension (PARAMETERS, SCIENTIFIC_* and CALIBRATION_DATE), because we have no standard procedure to apply scientific calibration yet. I want those variables and N_CALIBRATION to be optional.

Even those variables and N_CALIBRATION to be optional, the problem of missing values created in the process of the multi-profile file creation still remains. Should we fill them with some dummy values or define _FillValue attributes for all those values in format version 2.1?

Regarding missing profile data for the seven Japanese floats, the six, except for 4900293, stopped data transmission before JMA started real time data processing. 4900293 made no data transmission after launch. Japan can submit the data of the six floats, though no scientific QC completed as Dean pointed out.

From : Sylvie Pouliquen

Date : 13/12/2002

I don't know if you have taken a decision yet on this subject. If it is not the case I would advise you to keep these values and to put them at the min and max value for the paramater in the whole globe (ie for Temp valid_mins=-5 and valid_max=45). These values are usefull for people would want to make generic tools to plot a parameter and fix automaticall the scale of the axis.
A user working on a specific area will most of the time don't use it and fix his own scale but some does epecially with tools like Matlab and Idl.

If you are convinced by this argument, you should keep these fields and provide in the user manual the value that Dac should put in these fields

From : Thierry Carval

Date : 20/12/2002

Here is a proposal for data with no calibration.

When no calibration information is available :
. N_CALIB is set to 1
. SCIENTIFIC_CALIB_EQUATION, SCIENTIFIC_CALIB_COEFFICIENT, SCIENTIFIC_CALIB_COMMENT are set to "none"
. CALIBRATION_DATE is set to "00000000000000"

A Bob mentioned it, with NetCDF, the N_CALIB set to 0 would not mean empty calibration, but unlimited calibrations.

We also have a similar case with trajectories without measurement. Here is a proposal for this second issue :

Trajectories with no measurements
When no measurements are available on a trajectory, the temperature parameter is filled with fill values.

Is that correct ?

From : Charles Sun

Date : 20/12/2002 19:44:00

I would like to suggest that we don't include the calibration and history groups in the Argo NetCDF files, unless we have meaningful information or know how to handle "missing value" in netCDF files properly.
TC : I do not agree, these informations have to be consistently managed.

If we really like to have these two groups in the netCDF files, just Fill the fields with pre-defined "_FillValue", for example, "00000000000000" for CALIBRATION_DATE, but don't confused with the "missing_value" attribute.
TC : OK

The basic difference between the "_FillValue" and "missing_value" attributes is, for example, a netCDF file, XYZ, has a variable, X, with a attribute, "_FillValue = '99999'", then you will see "X = _," in the text presentation file created by the "ncdump" program. It means that we expect that Variable X should have a value but just does not have it in the file XYZ. If you see "X = '99999'" in the ncdump file, it means that we know it always does not have a value which we call it "missing value". Another example for "missing value" is the attribute values of the water temperatures variables over land points should be always "missing" (no water temperature observations on land points) and the attribute values of the water temperatures variables over water points can be "missing" and filled with "_FillValue" in netCDF files.

Hope that my description is clear. Please refer to http://www.bic.mni.mcgill.ca/software/minc/netcdf/guide.txn_59.html#SEC108 for additional information.

From : Bob Keeley

Date : 20/12/2002 20:14

I see no harm in what Thierry proposed. I suppose we could simply put blanks rather than the "none" Thierry suggests, but putting "none" shows there was a conscious choice of no information.

Regarding including groups only if meaningful, this doesn't sound attractive to me. First, if people are building netCDF files as we intend, everyone will be building at least one record in the history group. Allowing people to choose whether or not they write a calibration group would mean we would need to write code to bypass the software to read that group if no dimension was found. This just makes the software more complicated.
TC : I agree

From : Charles Sun

Date : 20/12/2002 22:21

To make myself clear, when I meant "meaningful" I meant that every variables should have "_FillValue" attributes filled with "meaningful" (standard) netCDF conventions. In other words, we can define the "_FillValue" attribute of the "CALIBRATION_DATE" has a value of "00000000000000" (14 zeros). If we do it correctly, the text representation of netCDF files should show CALIBRATION_DATE = _, ; (instead of CALIBRATION_DATE="00000000000000"). This is just an issue of the netCDF syntax. It may not be a big deal.
TC : OK, done

From : Charles Sun

Date : 22/12/2002 04:54

```
After carefully studied how characters are handled in netCDF,
I have convienced myself that Bob was right - we just simply put
blanks for all missing fields in the calib and history groups.
```

From : Bob Keeley
Date : 02/12/2002
Subject : additional QC information

From : Bob Keeley
Date : 02/12/2002
Subject : additional QC information

From : Bob Keeley
Date : 02/12/2002
Subject : additional QC information

From : Bob Keeley
Date : 02/12/2002
Subject : additional QC information

From : Bob Keeley
Date : 12/11/2002
Subject : additional QC information

```
Here is what I conclude from our emails. I have already taken steps to
incorporate those things I think are decided.

Decided
1. We have decided to insert fields <PARAM>_CORRECTED_ERROR in the D*.nc
files only. These will be Real*4 and contain the drift correction errors
generated from Annie's qc software. Thierry will need to update the Manual
to describe these fields and that they occur only in the delayed mode
files.
2. We have decided to modify the defintion of QC flags shown in table 3.2
to read as follows.
0 = no QC was performed
1 = good data: all tests passed
2 = probably good data: any inconsistencies or test failures are
considered insignificant after scientific evaluation against
local variability
```

3 = bad data that are potentially correctable: vertical profiles suggest significant jumps or drifts relative to initial instrument calibration or neighboring profiles, or significant deviations from climatology.
These data are not to be used without scientific correction
4 = bad data: data have failed one or more real-time QC tests
5 = value has been corrected by scientific methods
6 = not used
7 = not used
8 = interpolated value
9 = missing value. Used when original, bad values are replaced by fill_values.
3. We have decided that Thierry and Bob will rewrite table 3.7. Attached is
my proposal Thierry.
4. We have decided that Bob will write an explanation of how history structures are used. See the attached.
5. We have decided to refer to the AST the question of whether or not to record test failures at every level of a profile. See my attached question.

To be decided.
1. Do we use a flag of 5 for values changed as a result of calibrations or drift corrections?
2. If the AST requests we keep detailed test failures at each level, we need to decide how to do this.
3. Ann needs a code for the history structure to show there was a spike failure.
4. What is the correct way to write qc flags as arrays not long strings? Are there other fields like this to consider?

From : Ann Thresher
Date : 12/11/2002
Subject : additional QC information

Annie is right and that has been my hesitation about flag 5 as currently defined. How then do we indicate a change that hasn't been the result of a drift correction or scientific analysis? Looks like the flag on param_corrected becomes useless because it will always be 5 so why not just eliminate it? After all, the name tells you it's been corrected... I really think we need to revisit this one and use ONLY it to indicate actual quality of the data (1,2,3,4, 8 or 9 as currently proposed). This is much more useful information. Just because something has been scientifically corrected doesn't necessarily mean you trust it so 3 and 4 are still relevant. And 2 will also be used depending on how good you judge the correction to be. Or are we expecting the drift correction to result in perfect data? 5 as it stands just muddies the water.

From : Claudia Schmid
Date : 13/11/2002
Subject : additional QC information

> Decided
> 1. We have decided to insert fields <PARAM>_CORRECTED_ERROR in the D*.nc

Yes

> 2. We have decided to modify the defintion of QC flags shown in table 3.2

All pretty clear except for 5 as mentioned below.

> To be decided.
> 1. Do we use a flag of 5 for values changed as a result of calibrations or
> drift corrections?

Concerning the different opinions about 5. I agree that the 'changed value' information is not very important, since the original data are always availlable to find out about changes to data. Maybe eliminate it?

> 2. If the AST requests we keep detailed test failures at each level, we
> need to decide how to do this.

Yes, but let's wait for the AST.

> 3. Ann needs a code for the history structure to show there was a spike
> failure.

What about using 8 (interpolated value) or 9 (replaced with fill value). Removal of spikes typically requires to follow one of these two ways, and they are the only cases (what other cases can anyone imagine?) where such a traetment is necessary.

Comments about table ??? QC test IDs:
Platform ID: does not belong here.
This does not generate a flag. Problems with it only prevent data from a float to go to GTS, and it prevents the generation of a nc-file. We check if we ever had a used a newly assigned WMO ID before we use it for a new float (we sometimes were assigned the same ID twice). The problem I see is that we can not check what other DACs are getting.

Impossible date test: does not belong here, date is flagged in JULD_QC.

Impossible location, position on land and impossible speed test: do not belong here, they are flagged in POSITION_QC (together with date of position fix).

I'm still not sure what the difference between the digit roll-over test and the gradient test is. The former is much more lenient, but essentially a failure of the gradient test will occur whenever a digit roll-over occurs.
In addition: we are taking care of digit roll-overs in the decoding software.

Is there any chance that the top and bottom spike tests are coming back?

I still think we should stay away from the binary encoding.

Table `history software codes'
I don't have much of an oppinion about these.
But some of them seem weird to me: ARFM, IGO5 (relation to ARGQ???), IOC2, WOPF, QCA1 (relation to ARGQ???), QCAD (are there any standards about this).

Also: ARGQ, while performing the same basice tests they may still be not identical in the DACs. Also, what about the version numbers. Which changes require an increase of the version number and which don't?

For real-time profiles: If they have flags, then they went through the `ARGO QC' (or is it permitted that this is not the case?)

From : Ann Thresher
Date : 13/11/2002
Subject : additional QC information

Just a clarification of a misunderstanding by Claudia - we know what QC code we use for these missing values (9) - but we need a flag for the history records so people know WHY we replaced it. In woce, we use SP (spike) to signal that the action was taken because of a spike in the data. For Argo, we need a similar flag (either 1,2,3 or 4 characters will fit in the field) to justify the replacement. I haven't had a chance to look at the tables in detail and will do so soon...

From : Bob Keeley
Date : 13/11/2002
Subject : additional QC information

I was occupied with other things yesterday so didn't respond. However, that's good in that I get to see what everyone is thinking before responding.

Everyone agrees to the list of decided (except a precise definition of 5 which we will try to sort out today). So, we need the Manual to reflect this and to get on with implementing the things.

In another project using flag=5 made sense because there were corrections that could be made (in date and position, for example) where manual encoding of the data clearly caused things like reversed digits, lost signs on latitude and so on. We used 5 to indicate to a user that the value was changed should they wish to second guess the change. Of course, our operating principle was do not make a change unless you are quite certain you can improve the record.

In the Argo context, I am hard pressed to think of how a value might be changed in an analagous way. That is, I can only imagine changes as a result of drift corrections or calibrations. These tend to affect pretty near the whole profile.

A calibration tries to "re-align" observations to "more correct" values but until the data go through QC, the rightness of the calibrated values is untested. The passage of data through a calibration technique is recorded in the history. I would propose not to use any QC flags to indicate a calibration.

The passage of the data through the drift correction process is also recorded in the history and, as Annie pointed out, the field name in fact indicates changes are likely to have been made. I am nearly convinced to do as Annie suggested and simply not use 5. The only hesitation is that we may be overlooking some "ad hoc" process that results in a corrected value that I think we likely do want to flag. Of course, setting the flag 5 = not used now, does not cut off re-instating it at a later date. If you all are comfortable to do things this way, I agree as well.

Regarding Claudia's comments about QC test IDs, I would argue that whether or not a test is used by one group, as long as it is valuable to another, we can record it. She is right that Platform ID test does not result in a flag, and so I would agree to removing this from the list. I think the impossible date test is appropriate because it results in a flag being set in JULD_QC. The same comment applies to impossible location, location on land and impossible speed. These test IDs are to tell a user what tests have been performed / failed.

The drift rollover test is there because earlier models of floats had such problems. I do not think later models do. The gradient test looks similar to digit rollover, but so what? It would be nice to design tests that detect a single problem, but I don't know how to do this. The pragmatic solution is to design tests that look for peculiar oceanographic features and sometimes in different ways. I see no great harm in having two tests that are similar.

There is a chance top and bottom spike may come back, perhaps as different tests. They are gone for now until we can figure out a refinement that removes the undesireable qualities we found.

The HISTORY_QC test field is 16 characters and can hold 16 test results with no binary encoding. I respect Claudia's dislike of binary encoding, but don't agree. I think, though, that we should come to an agreement of how the encoding should be done.

The history software codes that you see are the ones that we employ. As I said, these can be defined however any group wishes to since the codes are associated with an organization. In most (all?) cases, no one but the organization creating the codes will care what they mean.

In summary, then.

1. We proceed to implement the "decideds" in my last email
2. I propose that calibrations do not result in any QC flag being set.
3. If others agree, we will change the defintion of qc flag=5 to "not used"
4. I think we should standardize on either a binary or not test reporting strategy.
5. History_software codes can be whatever is meaningful to an organization. It is their responsibility to document their meanings.

From : Claudia Schmid
Date : 13/11/2002
Subject : additional QC information

> In the Argo context, I am hard pressed to think of how a value might be
> changed in an analagous way. That is, I can only imagine changes as a
> result of drift corrections or calibrations. These tend to affect pretty
> near the whole profile.

I can think of a few cases where the position had to be changed. ARGOS is estimating two positions during a satellite overpass and then decides which one is correct. This decision is sometimes wrong (very rarely) and thus puts a position in the data file that is way off (e.g. +11.543 -73.119, versus +3.290 -37.359). We do not throw that position away (it will appear in the trajectory file with a flag of 4), but we prevent it from making it into the profile file. In my opinion this does not qualify as a changed value, because we use an original position supplied by service ARGOS.

This is the only time when we manipulate files manually, because the climatology QC can not be performed correctly if the profile position is wrong and because we don't want to transmit a profile like that to GTS etc... without the correction.

>   Regarding Claudia's comments about QC test IDs, I would argue that
> whether or not a test is used by one group, as long as it is valuable to
> another, we can record it. She is right that Platform ID test does not
> result in a flag, and so I would agree to removing this from the list. I
> think the impossible date test is appropriate because it results in a
flag
> being set in JULD_QC. The same comment applies to impossible location,
> location on land and impossible speed. These test IDs are to tell a user
> what tests have been performed / failed.

How can the binary encoding show if a test was passed or failed? The way
I understand it, it will only show if a test was done or not. Correct?

>   The drift rollover test is there because earlier models of floats had
> such problems. I do not think later models do. The gradient test looks
> similar to digit rollover, but so what? It would be nice to design tests
> that detect a single problem, but I don't know how to do this. The
> pragmatic solution is to design tests that look for peculiar
oceanographic
> features and sometimes in different ways. I see no great harm in having
two
> tests that are similar.

Digit roll-overs are float specific (i.e. the size of the jump depends on
the float type). That's why we correct them in the decoding software for
the individual float types.

> 1. We proceed to implement the "decideds" in my last email

agree.
Annie, when do you want us to switch to the new QC flags. We would
rather do it sooner than later. Is there a plan about when to release
the new data handbook?

> 2. I propose that calibrations do not result in any QC flag being set.

agree

> 3. If others agree, we will change the defintion of qc flag=5 to "not
used"

agree

> 4. I think we should standardize on either a binary or not test
reporting
> strategy.

we definitely have to agree on a standard

> 5. History_software codes can be whatever is meaningful to an
organization.
> It is their responsibility to document their meanings.

If this is so, then it may be nice to add the information about the
organization that suggested a code to the table?

From : Annie Wong
Date : 13/11/2002
Subject : additional QC information

Thanks for the clarification concerning the background behind flag=5.

Regarding the action of calibration, the existence/creation of a
D**.nc file should be sufficient to denote that calibration is
performed/available for a profile (if we can confidentally say
that nobody will go around creating D**.nc files when no calibration
is available!). I agree that a flag is not needed to denote this action.

Regarding the rightness/quality of the calibrated values, flag 5 as it is does not do anything here. People seem to think 1,2,3,4 are sufficient for now. Maybe in future when it comes time to passing calibrations to PIs for inspection, they will come back and request more. That's ok, especially when there're still 6,7 to play with.

However, I'm worried about the date and position examples that Bob cited. After reading Bob's document "Delayed_QC_Instructions.doc", on page 3, under Profile Data, point#2 for both JULD_QC and POSITION_QC, flag 5 is used as Bob has quoted in his examples. Do people think that maybe we should keep the usage of flag 5 as in Bob's example, but come up with some definition that specifies it is only manual adjustment to date and position, and irrelevant to profile_<param>s?

From : Ann Thresher
Date : 14/11/2002
Subject : additional QC information

First suggestion following Annie's pointing out that we still use a qc flag of 5 in some circumstances:

5 - value changed manually

This can then be used for anything - position, date/time or param - when it fits. And it eliminates all reference to scientific assessment or calibration or whatever.

Regarding Claudia's comment that binary encoding cannot show if a test was passed or failed, reading Bob's description of the history record shows how this field can tell what tests, if any, have been failed. And reading it carefully makes it clear how this is to be used. The only criticism I have is that "CV" is pretty useless as a History_Action. I guess it says that you changed the value but it doesn't tell anyone why it was necessary to do so. Maybe this can be covered in the HISTORY_QC_TEST field here where the justifying test is the only one set so, if you changed it because it failed a spike test, this field would hold the number 8192. this isn't very descriptive and I prefer a more blatant way to indicate why you took an action (like a code "SP" somewhere which is easily and intuitively decoded, but it works and all the information is there if you want to dig it out.

We still need the science team to tell us whether to store this information (my personal science team member thinks this is overkill but we'll see what the rest of the team thinks) so deciding where it goes is a bit academic at this point.

Which brings us back to overall data quality. I need to make sure I understand what is expected. First, for the R*.nc files, we put the raw data in the Param_corrected field if we haven't done a calibration and the param_corrected_qc becomes the overall QC associated with the real-time data and this is what is used by Annie. Correct? We put our "9" for a spike replaced by a missing value in the param_corrected_qc field. We have no way to flag the original, truly raw, value as bad because there isn't any QC flag associated with the param field. Suggestions how to deal with this since it really needs a QC flag of 4? This is relevant in the history fields where any change to the quality of the data, whether the raw or the corrected, needs to be recorded and, at present, only the corrected data is referenced in the history description. I should say that my personal opinion is that we need a param_qc field of a single character to store this sort of assessment. And, if we need to erect a variable to store Claudia's test failures (assuming we reject Bob's suggestion - though I think his solution is as good as any...) it should be named something clear like PARAM_TESTS_FAILED_QC, not PARAM_QC which already has connotations from param_corrected_qc... My two bits worth.

Regarding Thierry's comment about arrays/strings, this is why I went along with the majority, bowing to his greater knowledge of netcdf. The only way we can make ncdump create what looks like an array is to add a dimension "STRING1" to all QC character definitions. It works just fine and makes it much clear what you're talking about. But - then ncdump has many lines of single characters taking up your screen so, for looking at the data, I find I actually prefer the concatenated string representation. But I'm happy either way as long as we are all clear on it.

Finally, to Bob's points:

1) agreed
2) agreed!!
3) see above - suggest QC flag of 5 set to "changed manually"? since we have come up with some cases where it is still useful.
4) I prefer a system where only tests failed are reported and where the tests are spelled out (or abbreviated in some way) but this seems impossible... Binary works for me since I can ignore it completely once I set my software up to deal with it...
5) agreed.

Almost there?

From : Bob Keeley
Date : 14/11/2002
Subject : additional QC information

You are either being worn down or we are reaching agreement (or both). We are rapidly deciding the undecided. From the emails of yesterday I conclude:
1. There is some hesitation to discard use of flag 5, but Ann's suggestion of a defintion "value changed manually" seems to me to have the right sense. I can agree to this.
2. Ann's comment about wanting a more descriptive code in the history actions to explain why a value has been changed is permissable. This is what you did in GTSPP and I see no reason why you cannot do it here. You suggested using SP. Why don't you itemize and circulate your suggested action codes?
3. I will draft the question to the AST and circulate to you to ensure I got it right.
4. Thierry's argument about QC flag representation in netCDF sold me. In other emails we agreed to live with it (at least for now), so I am okay with this.
Is there further discussion or do the comments I made meet your approval?

From : Bob Keeley
Date : 14/11/2002
Subject : additional QC information

Sorry, I forgot to respond to Ann's summary of her understanding of what she is to do. Let me respond by my understanding.
In the R*.nc files we place the data in <PARAM> (where this is either temp or PSAL), and in <PARAM>_CORRECTED fields. The quality flags are put

in *_CORRECTED_QC. In Ann's case where she replaces a spike with a fill_value, the original value is in the <PARAM> array, the fill_value in the *_CORRECTED field and the QC flag of 9 in the *_CORRECTED_QC array. She will write a record in the history section using an appropriate action code saying the replaced value was a spike and placing the value of the spike in the _previous_value field.

I don't feel compelled to place QC flags against the original profile. In most cases, users will be content with the _CORRECTED profile since this is where all the work has been done to improve it. If they really want to go back to the original it is there and it is unlikely, I think that will will really care about any QC flags we may place there. If they are really interested, the information will be in the history records.

From : Claudia Schmid
Date : 14/11/2002
Subject : additional QC information

> Regarding Claudia's comment that binary encoding cannot show if a test was
> passed or failed, reading Bob's description of the history record shows how
> this field can tell what tests, if any, have been failed.

Yep, in the history record it's clear. When I wrote my comment I was thinking of the profile part not the history part (maybe PARAM_INDIVIDUAL_TESTS_QC is a good name). There we would have to have a way to encode three different possibilities (0, 1 and 4). This can not be done in the way described in the Table `QC test Ids'. If the AST says they don't need all those flags, then we can go either the `table' way or the binary way for the history. If the AST says we need all those flags, then the `table' way for both (history and individual) might be better.

I think that having a PARAM_QC with the original flags (in the same form as PARAM_CORRECTED_QC) would be good.

I had asked before what the $ stands for. I think now I know. I recommend for the history actions code table (if my interpretation is correct):

QCF$	Tests failed, the $ represents a value assigned as described in Table 'QC tests IDs'
QCP$	Tests passed, the $ represents a value assigned as described in Table 'QC tests IDs'

If a change of the profile time or position is done: why is START_PRES set to 0 and STOP_PRES set to fill value? Shouldn't both be set to fill value?

If a change to PARAM_CORRECTED is done: It also says set START_PRES to 0 and STOP_PRES to fill_value. Is this just because of `copy and paste'? It also says the previous value (I assume this refers to actual data and not to flags) is supposed to be stored. I'm not sure if this is always necessary.

If a whole profile goes through a drift correction, then the whole profile has to be placed in the history record? Considering that we still have the original profile in PARAM I think a drift correction does not require this. A different issue may be if a spike is being replaced after a drift correction (then the original value in PARAM is not the same as the one replaced).

From : Bob Keeley
Date : 14/11/2002
Subject : additional QC information

Below you will find the draft email I would send to the AST. Is this clear?
Regards, Bob

At the last Argo Data Management Team Meeting a question was raised concerning perserving detailed information about quality control tests. A few of us have been discussing this by email and have concluded we would like you advice on this matter.

At present there are a number of quality control tests that are run automatically on the real-time Argo profile data. We have also agreed to standardize the scientific quality control procedures to those proposed by Annie Wong. The discussion centres around how much detail is useful to keep about the results of the testing.

Presently, we record a data quality flag at each level in the profile. This flag states whether the observed value is good or bad or something in between. We also record that for a particular profile this or that test has failed. The discussion centred around whether or not we should be recording the individual test results at every level.

To be very clear, let's suppose we have three tests called A, B, C. What we record now is this
Tests performed = ABC
Tests failed = BC

Pressure	T	QC flag
Level 1	value 1	1
Level 2	value 2	3
Level 3	value 3	4
Level 4	value 4	1

Etc.

What we are discussing is adding another column for test failures at each level, so what we would record is:
Tests performed = ABC
Tests failed = BC

Pressure	T	QC flag	Failure
Level 1	value 1	1	
Level 2	value 2	3	C
Level 3	value 3	4	B
Level 4	value 4	1	

Etc.

We wish to have your advice on whether or not such detail will be used.

From : Claudia Schmid
Date : 14/11/2002
Subject : additional QC information

Bob,

Thanks for formulating the letter.

Maybe use
Level 3 value 3 4 BC
in the example?

> level, so what we would record is:

```
> Tests performed = ABC
> Tests failed = BC
> Pressure     T       QC flag  Failure
> Level 1   value 1       1
> Level 2   value 2       3       C
> Level 3   value 3       4       B
> Level 4   value 4       1
> Etc.
```

I realize that I'm fighting a loosing 'battle', but let me say this:

An absence of A can mean 'A was not done' or 'A was passed'. This makes it necessary to also read the history record to find out if A was performed. This is why I'm not in favor of the binary way. A completely unique way of assigning the values at each level is safer (and more user friendly) than one where additional information is needed to correctly decode the flags.

From : Ann Thresher
Date : 14/11/2002
Subject : additional QC information

Comments in caps as usual, Ann.

-----Original Message-----
From: Claudia Schmid [mailto:schmid@aoml.noaa.gov]
Sent: Friday, 15 November 2002 2:37 AM
To: Ann.Thresher@csiro.au
Cc: Bob Keeley; awong@pmel.noaa.gov; Thierry Carval
Subject: RE: additional QC information

Dear all,

> Regarding Claudia's comment that binary encoding cannot show if a test was
> passed or failed, reading Bob's description of the history record shows how
> this field can tell what tests, if any, have been failed.

Yep, in the history record it's clear. When I wrote my comment I was thinking of the profile part not the history part (maybe PARAM_INDIVIDUAL_TESTS_QC is a good name). There we would have to have a way to encode three different possibilities (0, 1 and 4). This can not be done in the way described in the Table `QC test Ids'. If the AST says they don't need all those flags, then we can go either the `table' way or the binary way for the history. If the AST says we need all those flags, then the `table' way for both (history and individual) might be better.

GOOD PLAN..

I think that having a PARAM_QC with the original flags (in the same form as PARAM_CORRECTED_QC) would be good.

I THINK BOB HAS THIS RIGHT AND WE CAN IGNORE A PARAM_QC FIELD.

I had asked before what the $ stands for. I think now I know. I recommend for the history actions code table (if my interpretation is correct):

```
QCF$   Tests failed, the $ represents a value assigned as described
       in Table 'QC tests IDs'
```

QCP$ Tests passed, the $ represents a value assigned as described in Table 'QC tests IDs'

I'M NOT SURE YOU HAVE THIS RIGHT BUT WILL LEAVE THIS TO BOB TO DEAL WITH. I DON'T THINK THE $ IS REPLACED BY ANYTHING. AM I RIGHT? THE VALUE CLAUDIA CITES GOES INTO THE HISTORY_QC_TEST FIELD. OR AM I INTERPRETING CLAUDIA WRONG?

If a change of the profile time or position is done: why is START_PRES set to 0 and STOP_PRES set to fill value? Shouldn't both be set to fill value?

NO COMMENT

If a change to PARAM_CORRECTED is done: It also says set START_PRES to 0 and STOP_PRES to fill_value. Is this just because of `copy and paste'? It also says the previous value (I assume this refers to actual data and not to flags) is supposed to be stored. I'm not sure if this is always necessary.

AGREED - START_PRES SHOULD BE THE POINT OF THE ERROR AND STOP_PRES SHOULD BE THE END POINT OF THE ERROR. USUALLY THIS WILL APPLY TO ONE POINT ONLY (IF SOMETHING HAS CHANGED, EACH CHANGED POINT REQUIRES ITS OWN HISTORY RECORD SO THE ORIGINAL VALUES CAN BE PRESERVED) SO START AND STOP WILL BE IDENTICAL.

If a whole profile goes through a drift correction, then the whole profile has to be placed in the history record? Considering that we still have the original profile in PARAM I think a drift correction does not require this. A different issue may be if a spike is being replaced after a drift correction (then the original value in PARAM is not the same as the one replaced).

YES - WE DECIDED WE DIDN'T NEED TO FLAG THE DRIFT CORRECTIONS. BUT DO WE NEED A HISTORY SOFTWARE CODE FOR THE DRIFT CORRECTIONS? THIS WOULD THEN APPLY TO THE WHOLE PROFILE RECORD AND BE ONE HISTORY RECORD. ONLY IF A SPIKE OR WHATEVER CHANGES A VALUE DO WE NEED TO HAVE A HISTORY RECORD PERTAINING TO A DATA POINT THAT HAS BEEN DRIFT CORRECTED, AS YOU STATED ABOVE.

From : Annie Wong
Date : 14/11/2002
Subject : additional QC information

I have no comment on your draft letter to the AST.

I have no comment on "5 = value changed manually".

Regarding Thierry's comment about arrays/strings, I finally went to Matlab to try see what the confusion is. In Matlab the qc flags appear correctly as an array with the same dimension as the associated parameter. The definition is fine. Everything looks fine. I can't see any confusion. So I have no comment on this either.

And no, please, don't put whole profile in the HISTORY record after drift correction ... the original can be found in PARAM, and the purpose of having D**.nc files is for this record. Also, I'm not sure if we need a HISTORY SOFTWARE CODE for drift correction. Again, the purpose of having D**.nc files is for this record, and, when calibration is available and a D**.nc file is created, the

```
writer has to fill in the CALIB_ section, which contains all
calibration info.
```

From : Ann Thresher
Date : 14/11/2002
Subject : additional QC information

```
> -----Original Message-----
> From: Bob Keeley [mailto:keeley@meds-sdmm.dfo-mpo.gc.ca]
> Sent: Friday, 15 November 2002 3:29 AM
> To: 'Ann.Thresher@csiro.au'; Claudia.Schmid@noaa.gov
> Cc: awong@pmel.noaa.gov; Thierry.Carval@ifremer.fr;
> 'Pouliquen, Sylvie'
> Subject: RE: additional QC information
>
>
> Dear All,
>    Below you will find the draft email I would send to the
> AST. Is this
> clear?
> Regards, Bob
>
> At the last Argo Data Management Team Meeting a question was raised
> concerning perserving detailed information about quality

concerning pREserving detailed information about quality

> control tests. A
> few of us have been discussing this by email and have
> concluded we would
> like you advice on this matter.

like youR advice on this matter.

>    At present there are a number of quality control tests
> that are run
> automatically on the real-time Argo profile data. We have
> also agreed to
> standardize the scientific quality control procedures to
> those proposed by
> Annie Wong. The discussion centres around how much detail is
> useful to keep
> about the results of the testing.
>    Presently, we record a data quality flag at each level in
> the profile.
> This flag states whether the observed value is good or bad or
> something in
> between. We also record that for a particular profile this or

We also record ELSEWHERE that for a particular profile this or

> that test has
> failed. The discussion centred around whether or not we
> should be recording
> the individual test results at every level.
>    To be very clear, let's suppose we have three tests called
> A, B, C. What
> we record now is this
> Tests performed = ABC
> Tests failed = BC
> Pressure     T       QC flag
> Level 1   value 1       1
> Level 2   value 2       3
> Level 3   value 3       4
> Level 4   value 4       1
> Etc.
>
> What we are discussing is adding another column for test
> failures at each
> level, so what we would record is:
> Tests performed = ABC
> Tests failed = BC
> Pressure     T       QC flag  Failure
> Level 1   value 1        1
> Level 2   value 2        3       C
> Level 3   value 3        4       B
> Level 4   value 4        1
> Etc.
```

THIS IS A GOOD GENERAL DESCRIPTION AND WE CAN DECIDE ON DETAILS LATER IF THEY WANT THE INFORMATION. SHOULD WE INDICATE THAT THE FAILURE FIELD MIGHT
BE SEVERAL CHARACTERS LONG IN CASE A VALUE FAILED MORE THAN ONE TEST?

From : Ann Thresher
Date : 15/11/2002
Subject : additional QC information

Before I start coming up with an exhaustive (?) list of action codes for the history records, I need to clarify the use of some of the codes Bob has suggested. In particular, what are the differences between the use of CV (change a value), ED (edit a value) and SV (set a value). I assume there is some reason to have all 3?

Actually, I find codes like these relatively useless since they don't tell you why the action was taken. This includes the code of CF for change flag since that would require two history records for every quality change – one to record that you changed a quality flag and the other to record why....
The flag change should be obvious and not need to be indicated individually or we risk multiplying our history records with no added information... Any other opinions?

Also, if anyone would like to send me examples of failures they have seen that should be flagged with a history record, (i.e., that result in a change to the parameter or its flag in some way) I would appreciate it. I would like to cover as many bases as possible so we don't need to revisit this again.

From : Ann Thresher
Date : 15/11/2002
Subject : additional QC information

OK - maybe we should get radical. I've just been giving a lot of thought to the difference between the tests failed (or run or passed) and the reasons why we change a flag or data value. They are really the same thing. SO -
if we redo the way the history record is used, we might be able to solve

everyone's problems (except maybe Bob's preference for a binary test id).....

If you decide this is unworkable, fine....

I propose we keep CV and CF for the history action field. The big change is that we now create a new history record for every test run. We have room since N_HISTORY is unlimited. This shouldn't create a big overhead since we only have a double handful of tests at the moment and I can't see us coming up with a lot more in the near future...

We give each test a simple and descriptive name ('Spike test', or 'Climatology test' or 'Date test' or 'Location test') and store this in the History_QCTEST field. Since we have 16 characters to play with, this should be easy. And this satisfies both Claudia's unhappiness with binary coding and my desire for intuitively simple descriptions of what we have done to the data.

If a cast or parameter passed a test, the History_action becomes QCP$ to indicate that this particular test was run and passed.

If it fails the test and the test results in no change to the data, the History_Action field becomes QCF$ and either the GF3 code goes in the History_parameter field or we use RCRD to indicate it applies to the entire record.

If the failure results in a change to the data, and the entire parameter or record is affected, it is indicated by a History_Action field that contains either a CV or CF and the start and end pressures indicate the entire record.

If part (but not all) of the profile passes the test (i.e., spike test) then the RCRD gets a history that says this test was run - QCT$ (QC test) or QCR$ (QC run) to be consistent. The part that failed requires an additional history record that gives the start and stop pressures of the spike (or area that lies outside climatology) and indicates that either the value was changed (CV - if you interpolated or replaced by missing) or the flag was changed (CF - if you merely set the QC flag to 4 - or 3) in the history action field. In all cases, if you change the data, you needn't indicate that you also changed the flag since this is implicit. So at most you get one additional history record for a failure. It will have to be implicit that any part of the profile not explicitly referenced as a failure of this test was, by default, a pass. That is easy to understand and communicate though you wouldn't know it from that last sentence....

Additionally, if this is used right, it also gives some indication of the test results per profile data point. If a test was failed, the depth ranges or the failure are shown in the relevant history record and the parameter is identified in the History_Parameter field. If a test was passed, it is implicit that all levels passed and we don't need a variable or field to flag each one. Perhaps this is a reasonable compromise.

So - if we do this, we make it easy for anyone interested to read the tests done and decipher which data values failed which test. No decoding or formulas or look-up tables. And we don't have to come up with new history action codes since they are covered by the test definitions. If we add 15 or 20 records to the history fields, or even 30 records (in the worst case where it fails almost everything), so what? You only look at them if you're keen.

I think this will work well but I've probably missed something or haven't made it clear ... Feedback?

From : Bob Keeley
Date : 15/11/2002
Subject : additional QC information

Now to address the history things that came up yesterday.

First off, it seems Claudia misunderstood the $ in QCP$, QCF$. The $ is not a wildcard character, it is a "$" only. The $ does not get replaced by anything. When we first started using 4 character codes we adopted GF3 forms. We found that we needed other ones and to keep straight which were new we decided to always set the 4th character to be a $. That is its only significance.

Setting Start_Pres to 0 and Stop_Pres to fill_value is simply what I thought was sensible. The station level things do apply to the surface, sort of, and so I said set to 0. I am happy to simply set to fill_value if it makes more sense to the others.

If a value is changed, the philosophy is to record the previous value before the change. For changes to date, lat, long, this is the only way we can preserve what was there before. In profiles, this may not be necessary if we do not care about all of the corrections along the way, simply the original and the last, best version. I would like to hear opinions on this.

Just as we apparently agreed that we need not set the flag of 5 on all levels when a drift correction is made, I do not think we need record previous values from all levels in the history section.

I did not understand Claudia's question "...what is the purpose of QC?"

Annie remarks that we do not need to have a history software code to indicate data have gone through scientific QC. I disagree. Despite our best intentions, I can image a D*.nc file being created that has not seen scientific QC. I feel it is important to explicitely declare this process than infer it from a file name. Concerning calibration, Annie is quite right that when a float goes through calibration, the appropriate information should be in the calibration file. Though it seems redundant to have a record in the history to say the data went through calibration, too, it is a small thing and puts all of the processing information in one place. I can be convinced this is unneccessary if others agree with Annie.

I hope this covers everyone's comments.

From : Claudia Schmid
Date : 15/11/2002
Subject : additional QC information

Sounds to me like QCF$, QCP$ are a widely used standard. So let's keep them. Table 3.7 has many two character codes. Is there any intention to add $$ to them?

> Setting Start_Pres to 0 and Stop_Pres to fill_value is simply what I thought was sensible. The station level things do apply to the surface, sort of, and so I said set to 0. I am happy to simply set to fill_value if it makes more sense to the others.

Changes in position and/or time are valid for the whole profile, but no changes are done within the profile data. So I think anything other than a fill value may be misleading.

> If a value is changed, the philosophy is to record the previous value

> before the change. For changes to date, lat, long, this is the only way we
> can preserve what was there before. In profiles, this may not be necessary
> if we do not care about all of the corrections along the way, simply the
> original and the last, best version. I would like to hear opinions on this.

My opinion is: If a position is definitely wrong (like in the example I gave previously), then we can not perform the QC properly without preventing the software from using the erroneous position/time record. I don't see much value in retaining such a position and the associated time in the history of the profile file. If we are to put the erroneous position/time into the history, then it has to be made clear that the position/time were only replaced with a position/time reported by service ARGOS at a later time, i.e. they were not changed by hand-editing or scientific methods (like extrapolation).

> I did not understand Claudia's question "...what is the purpose of QC?"

If a QC is performed the HISTORY_ACTION is either QCP$ or QCF$. I can't think of a case where a QC is performed and the HISTORY_ACTION is set to QC.
If all agree I suggest for the table:
remove the line with QC

change the lines defining QCF$, QCP$ to:
QCF$ Quality Control tests Failed
QCP$ Quality Control tests Perfomed

From : Annie Wong
Date : 15/11/2002
Subject : additional QC information

No comment on second draft of letter to AST.

>If a value is changed, the philosophy is to record the previous value
>before the change. For changes to date, lat, long, this is the only way
>we can preserve what was there before. In profiles, this may not be
>necessary if we do not care about all of the corrections along the way,
>simply the original and the last, best version. I would like to hear
>opinions on this.

For profile params, in R**.nc files, most changes are going to be simple and the current set-up of param + param_corrected is sufficient. In D**.nc files, the scientist who judges the quality of the corrections may want to preserve previous values. He/she has the option to do so in the current set-up in the history section.

>Just as we apparently agreed that we need not set the flag of 5 on all
>levels when a drift correction is made, I do not think we need record
>previous values from all levels in the history section.

Yes, no flags, no record ... in this case because there are no previous values. Original measurements stay in <PARAM>.

>Annie remarks that we do not need to have a history software code to
>indicate data have gone through scientific QC. I disagree. Despite our
>best intentions, I can image a D*.nc file being created that has not seen

>scientific QC. I feel it is important to explicitely declare this process
>than infer it from a file name.

Then the way to go is to insist D**.nc file creators put in a history section but allow them to come up with their own software codes. That provides more specific info than a general declaration that scientific corrections exist. Bob has said in a previous email that the history software codes can be whatever is meaningful to an organisation.

I have a question ... does this mean the history section is compulsory? The current R**.nc files do not have a history section. Before this discussion I had simply assumed that that meant the file had been created with no subsequent history to report. It's obvious that the data had been through real-time qc by the fact there were qc flags. Following that logic, I then assumed that when I created a D**.nc file, I would put in the calibration info ... it would then be obvious that the values I put in <param>_corrected were calibrated. There was no history to report, so there's no history section, until the file went to the scientist for examination, then there'd be history and that's when the history section would be written for the D**.nc files.

Does the current discussion on various history codes mean that the "creation" of a file immediately entail a history section?

I am away next week in the WOCE conference and will be out of email contact. Will try to catch up the week after that. Thanks all,

From : Bob Keeley
Date : 18/11/2002
Subject : additional QC information

If I do not receive any further comments on the letter to the AST, I'll send it along to the AST by mid week.

Annie notes that we should insist D*.nc files have history sections. In fact, the R*.nc should as well since this is where the tests performed and tests failed should be written. Granted we are all supposed to be doing the same testing, but we know that not all of the data coming through Service Argos, for example, goes through tests. In fact, MEDS R*.nc files have history records, since this is standard procedure here.

The intent of the history section is to have a place to record "significant events" in the processing of the data. I consider calibration to be a significant event and so would like to see people write a record when this process happens. At MEDS we record most every event in the processing because we have found that it helps us sort out problems when they arise because we have the sequence of events that the data passed through.

A simple file creation is a significant event, but the fact that the file is present attests to it having happened.

In response to Claudia's questions, I do not feel compelled to add $$ to the two character codes to make them four characters. We could if people insist.

I agree with her commnts about positions and times that are definitely wrong. However, it may be that they are not thought wrong until more work is done on the trajectory data. Also, if this results in a correction by someone, another user may want to question the correction, so having the original there allows for that.

Responding to Ann's questions, I do not feel compelled to write 2 history records, one to indicate a changed flag and one to indicate a changed value, when a value is changed. The former action results from the letter.

I concede that CV and ED pretty much say the same thing. There are a

few codes that indicate some reasons, but not many. We should perhaps do
better and add some more if you think you would like to.

From : Claudia Schmid
Date : 18/11/2002
Subject : additional QC information

> In response to Claudia's questions, I do not feel compelled to add $$
to
> the two character codes to make them four characters. We could if people
> insist.

Great, I think adding them would only cause confusion.

> I agree with her commnts about positions and times that are
definitely
> wrong. However, it may be that they are not thought wrong until more
work
> is done on the trajectory data. Also, if this results in a correction by
> someone, another user may want to question the correction, so having the
> original there allows for that.

OK, so you want a history record for the given example (one position
hundreds of kilometers away from all other positions).
I don't think it is appropriate to give a POSITION_QC of 5, because we did
not change a position or time derived by service ARGOS. We only ignore the
first one when we generate the profile file if it is definitely wrong. The
way we identify the bad positions is by changing the ARGOS location class
to a negative number. We don't use a software for this so we will have to
leave the HISTORY_SOFTWARE... fields open (or set them to MANUAL, 0.0 or
so). The HISTORY_DATE is basically the date when the profile was first
received (not if it happens on a weekend). Since this case involves time
and position we need three history records.

From : Bob Keeley
Date : 18/11/2002
Subject : additional QC information

Dear Claudia,
I am not sure I understood everything you said. If you have a position
associated with the profile that is very wrong, what do you do?
If the same is true for the trajectory data, I assume you flag the
position as wrong. The same applies to time. If you modify the position,
then you need to write a history record saying the position was changed
and record what it was before the change.
I don't think you should be modifying the Argos location class since it
is what it is.

From : Claudia Schmid
Date : 18/11/2002
Subject : additional QC information

Bob,

> I am not sure I understood everything you said. If you have a
position
> associated with the profile that is very wrong, what do you do?

Some background info:
Our internal format did not include a seperate flag for the position. This did not seem necessary when the format was developed (in 1997 ... the first time the problem occured was in 11/2001). In our internal format we solved the problem by setting location class 1 to -1 (...). This change is reversible, and has the advantage that it did not require reprocessing all data obtained over more than 4 years. It was also a good solution because we could apply it in the hex files (the raw data of each float are stored in these files) without changing the format of these files. Therefore it also saved us from changing the decoding software for about 10 float formats.

trajectory files:
In the netcdf trajectory file we go back to the original value of 1 (...) and flag the position as bad. We do not flag the time as wrong, because it is the correct time.

profile files:
In the profile file we skip over a position with a negative location class. I would not use the term `modify the position' for this because that's not what we are doing. We are only ignoring a position and the associated time which should have been removed by service ARGOS (in 99.9% of the cases they do it, but occasionally a bad position slips through).

I talked with ARGOS about our 4 cases, and they explained to me that their algorithm derives two position estimates for each satellite overpass and then is supposed to pick the correct one (or none if the fixes are too bad). I showed them our 4 cases. They went back to the rawest data they have, which showed that the selction process had failed by picking the wrong position.

If you have any more questions about this please call me.
I think talking may be more efficient than email in this case.

From : Bob Keeley
Date : 19/11/2002
Subject : additional QC information

Dear Claudia,

I understand now what the location class discussion is about. What you explained sounds fine. As long as the netCDF files that appear at the GDAC conform, and information is not lost or rearranged so that others cannot understand, I am happy. How you run things internally is your business.

From : Annie Wong
Date : 28/11/2002
Subject : additional QC information

I returned from the WOCE conference with some feedback, and some new perspectives.

I've witnessed two incidents where a scientist downloaded data from the Argo site, plotted up a profile and found gross outliers. Reason ... they did not read the qc flags.

This says to me that no matter how many documents and handbooks there are out there, no matter how much effort we put into writing the history section, some people are just not going to read them. There is another problem that has been at the back of my mind but is now becoming clearer ....

... the delayed-mode data stream relies 100% on the real-time data stream. Delayed-mode calculations will always be based on the "good" raw values in <param>, so will always need to read qc flags associated with <param>. The only qc flags that are available now are in <param_corrected_qc>, which pertain to <param_corrected>, which, so far, by default, provide the qc information for <param>, because, so far, in most cases, <param_corrected> = <param>. This system will collapse when real-time correction becomes available, when <param_corrected> will have values other than <param>. At that time, qc flags associated with <param> will be lost, or buried in the history section. A bulk processing system like the delayed-mode data stream will not be able to decipher these one by one.

I'm hoping this panel will reconsider a few points that have been raised by Ann and Claudia previously:

1). Put in <param_qc> (not as a GF3 code please), using the flags in Table 3.2 whose definition we've agreed on. <param> and <param_qc> will never be altered by any process. With this in place, any future developement in all other fields/files/streams/flags/whatever, can evolve independently without affecting the easy availability of these primary information that all processes are based on.

2). Use the <_corrected> fields to make the data more fool-proof by adopting CSIRO's practice of replacing bad data with missing values in <param_corrected>.

My apologies for digging these things up again.

From : Bob Keeley
Date : 28/11/2002
Subject : additional QC information

First off, Annie, thanks for the email telling us what you saw. My first reaction was to find it depressing that people who read the file can't be bothered to look at the QC flags. I hope you pointed out to them their omission.

Regarding Annie's suggestions let me respond by telling you what we are doing and see if that lines up with what others do. When we write the real-time files, we write the data to both <param> and <param>_corrected and use the <param>_corrected_qc to store the qc flag resulting from the auto qc. As the data pass through more QC, the input to the process should be what is in <param>_corrected and respect the qc flags to make corrections and so on in this field. As Annie said, we leave the <param> untouched. So, anyone reading the data file should always look to the <param>_corrected fields and the associated flags. If people ignore the flags, they get what Annie saw.

The down side of doing as Annie suggested is that there will always be instances where what someone regards as bad, someone else does not - those borderline cases. If we replace the bad value with a missing value and place the previous value in the history, this makes it more difficult to recover the value to decide for yourself if the value is borderline good or bad. This is exactly the place where using QC flags helps, but only if people respect them.

Should we ask the AST for quidance on this? Annie, were the two examples you saw exceptions or was their a general sentiment to present only what we consider "clean" data in the <param>_corrected fields?

Regarding Ann's further question about one history for each test performed / failed, I don't favour it. This is another variation of a simplified coding of this information. Personally I would rather use Claudia's string coding that puts all information in a single history record, if the binary encoding is unacceptable.

From : Claudia Schmid
Date : 29/11/2002
Subject : additional QC information

did the scientists look at <PARAM> or <PARAM>_CORRECTED? If they read PARAM they were doing what most people would probably do first when getting data from a new source (it's just more intuitive). In this case they won't find any flags. This is, of course, no excuse, but I think it's likely to happen to many new users.

Maybe a way to reduce the probability that users make this mistake is to put a README file on the ftp sites, that tells the users the most important things about the data. They are much more likely to look at a 1-2 page ASCII document than at the user manual that looks more like a handbook for somebody who wants to create netcdf files, or who wants to dig deeply into the history record (which was the initial purpose of this handbook).

The most important items for such a README file are:

- inform the users about the presence of the _CORRECTED fields.
- inform users about the difference between R*.nc and D*.nc files
- supply the users with the information about the meaning of the QC flags (and the naming convention "_QC").
- tell them where they can get the user manual if they need more information.
- more?

If you agree with the above I can start writing such a README file and pass it around for comments and additions.

I fully agree with Annie about the delayed mode QC necessities. I think it is important to introduce <PARAM>_QC as a standard variable (not as a GF3 code) in the netcdf file.

Concerning replacing bad data with fill values: If this is done it should be only done for those that are really bad, like constant value profiles or data points that are out of range. With spikes I see a problem, because small spikes may be fine whereas big spikes are definitely bad, i.e. the decision between 'keep' or 'set to fill value' is somewhat subjective.

From : Annie Wong
Date : 29/11/2002
Subject : additional QC information

1). One of the proposed actions resulting from the Ottawa meeting is that in the next year or so, delayed-mode centres should aim to provide projected drift corrections to DAC's in real-time. When that step becomes a reality, the projected corrections will go into <param_corrected> in the R**.nc files. I'm not clear whether the flags in <param_corrected_qc> will be changed accordingly? I understand

<param> will never change. What I need are the qc flags that go with <param> that also will not change. If <param_corrected_qc> serves that purpose, then its relation to <param> is not clear. As Claudia said, if a person is reading <param> and does not see a variable called <param_qc>, he/she won't find/read the flags. That is my worry ... that a lot of these information are there but are only implicit or buried, whereas a simple <param_qc> that goes with <param> is logical, intuitive, clear as daylight, does not require any in-depth knowledge of the Argo data system or looking at any documents.

2). I don't know if there's a general sentiment to present only "clean" data in the <param_corrected> fields, but CSIRO's practice of replacing bad values with missing values did immediately pop into my head as a good way to fool-proof things, hence my suggestion. What I refer to as "bad" are the grossly out of range values ... totally absolutely deterministically "bad", like 30C water down at 1000 m!! No borderline. <param_qc> will be 4. <param_corrected_qc> will be 5. People won't have to dig up the history to check previous values ... they can just look at <param>. But as Bob said before, how different centres run things internally is up to individual centres. Something like this can may be suggested to various real-time centres as an option if it appeals to them. Again, I think if <param> and <param_qc> were in place, people would be less timid to experiment with <param_corrected>, because then they know the primary info are alway going to be there.

From : Claudia Schmid
Date : 02/12/2002
Subject : additional QC information

> 1). One of the proposed actions resulting from the Ottawa meeting
> is that in the next year or so, delayed-mode centres should aim to
> provide projected drift corrections to DAC's in real-time. When that
> step becomes a reality, the projected corrections will go into
> <param_corrected> in the R**.nc files.

We did not make a final decision on the QC procedures for profiles that go through a drift correction in real-time yet. My first plan was to only perform a QC of the drift corrected data. This would not change the outcome of the spike and gradient tests, but the other tests, primarily the statistical ones, may result in different flags. If we want to have a PARAM_QC (as you know: I do) then we will have to do the QC tests twice.

From : Bob Keeley
Date : 02/12/2002
Subject : additional QC information

I have no problems with the "totally, absolutely, deterministically bad", it is the "maybe, possibly, close to sensible bad" that gives me grief. But more to the point. Annie and others would like a <param>, <param>_qc, <param>_corrected and <param>_corrected_qc. We store the original values as returned initially from the float in <param>. Any corrections that might be done, or the best version of the profile is found in <param>_corrected.

Annie makes the point that she needs (the scientific qc process?) the original qc flags attached to the <param>. If everyone agrees, let's add <param>_qc. This would apply to profile and trajectory files, I presume.

The point of whether to substitute a missing value for bad values in the <param>_corrected profile is a separate issue. Personally I do not like this, since I don't like making calls on marginally bad data, as

Claudia noted. That is exactly why we choose to set QC flags. It seems the point of inserting the missing value is to "help" users who do not choose to look at the QC flags. I would advocate educating them, rather than "dumbing down" the data system. I will listen to your collective wisdom.

From : Bob Keeley
Date : 02/12/2002
Subject : additional QC information

I have no problems with the "totally, absolutely, deterministically bad", it is the "maybe, possibly, close to sensible bad" that gives me grief. But more to the point. Annie and others would like a <param>, <param>_qc, <param>_corrected and <param>_corrected_qc. We store the original values as returned initially from the float in <param>. Any corrections that might be done, or the best version of the profile is found in <param>_corrected.

Annie makes the point that she needs (the scientific qc process?) the original qc flags attached to the <param>. If everyone agrees, let's add <param>_qc. This would apply to profile and trajectory files, I presume.

The point of whether to substitute a missing value for bad values in the <param>_corrected profile is a separate issue. Personally I do not like this, since I don't like making calls on marginally bad data, as

Claudia noted. That is exactly why we choose to set QC flags. It seems the point of inserting the missing value is to "help" users who do not choose to look at the QC flags. I would advocate educating them, rather than "dumbing down" the data system. I will listen to your collective wisdom.

From : Bob Keeley
Date : 07/01/2003
Subject : additional QC information

I have been going through all the notes and this is what I think was decided.

1. We need to add fields PRES_QC, TEMP_QC, <PARAM>_QC. We would store the original profiles in * and original QC flags in *_QC. As corrections are made, or calibrations carried out, the results go into *_CORRECTED and flags in *_CORRECTED_QC. I can see using *_CORRECTED_QC to store flags of 5 for changed values. So where we can correct an individual level value, we would store the original in * and the original QC flag in *_QC (presumably a flag value of 4). Then, the corrected value goes in *_CORRECTED and the flag of 5 goes in *_CORRECTED_QC. In this case, the *_CORRECTED field contains the best values but a user is alerted if some change has been made by the presence of the 5 as a QC flag. We agreed, I think, that if the scientific QC process adjusted many or all values in a profile, we would not mark a 5 against them all.
2. In the comment field for <PARAM>_CORRECTED I think the original wording of the Users Manual (dated 16 July, 2002) is the right one.
3. In the comment field for <PARAM>_CORRECTED_ERROR, I suggest the comments read "<PARAM>_CORRECTED_ERROR contains the error on the corrected values as determined by the scientific QC process. This field is not mandatory if scientific QC has not been applied."
4. Reference Table 2 (for QC flag definitions) looks right.
5. Reference Table 7 looks okay.
6. Reference Table 3 looks much better. However, I think the valid limits on PSAL should be 0 to 42 and for TEMP should be -2 to 40.
7. I am of two minds about reference table 12. On the one hand, it would be nice to standardize the events people record in the history. On the other hand, I am not sure we can. What you assembled is a mixture of what MEDS currently uses and additional ones that look to be useful. What I

suggest is this. We construct a table with suggested important events to be recorded in the history section, but include a note that says if individual centres wish to record others, they may add to this list as they feel is appropriate.
Here is my suggested list:
ARFM - Convert raw data from telecommunications system to a processing format
ARGQ - Automatic QC of data reported in real-time has been performed
IGO3 - Checking for duplicates has been performed
ARSQ - Scientific QC has been performed
ARCA - Calibration has been performed
ARUP - Real-time data have been archived locally and sent to GDACs
ARDU - Delayed data have been archived locally and sent to GDACs
8. Concerning your comment/question 2, the calibration section has a date so the "when" is covered. I believe that the "how" is represented by the information stored in the calibration section. We can record the "who" by insisting everyone write a history record with the code ARCA as described in table 12. Since this is associated with who wrote this record and the date, we can associate records in the calibration section with who did it.
9. We heard that having valid maxs and mins was useful so I think these need to be present. Your comment is quite right about such broad ranges, but if one is plotting a variety of profiles, the full scale may be better used.
10. The format statements I think are there to meet standards for netCDF. I note that in the text you sent to me, there was a space between _CORRECTED and the :. This is also true in the _ERROR as well. I think this is a mistake.
11. I think including a comment is helpful. It gives us a chance to be more descriptive of what a particular field holds.

I have only had two replies regarding storing test failures at each level in a profile. One wanted this and one did not. If we should do this, it means we would need another field. However, lets consolidate what you have now, and send this out to the others who took part in the discussion. I will raise this topic separately.
I will be unavailable until next Tuesday, so make the changes discussed above and get it out to people for final comment. Thanks for all your help on this.

From : Annie Wong
Date : 10/01/2003
Subject : additional QC information

Thanks for the update to the User Manual.
It all looks fine to me, except for on the first page of your note, in the 3rd column under "Comment", I would use the word "variable" instead of "dimension". E.g ...

"This [VARIABLE] is not mandatory if no correction is available."

for <PARAM>_CORRECTED, <PARAM>_CORRECTED_QC, <PARAM>_CORRECTED_ERROR.

From : Claudia Schmid
Date : 10/01/2003
Subject : additional QC information

Here are my suggestions for table 3:

CNDC valid min 0 valid max 20 %10.4f F10.4 0.0001f

```
DOXY   valid max 550               %10.2f   F10.2   0.01f (or 10.3 0.001)
PSAL   valid max 60 (as currently)
TEMP   valid min -2.5 (as currently)
```

Suggestions/comments for table 11:

What is the purpose of 'Platform Identification' (without a positive identification we can not create the netcdf files)? Removing it leaves 13 tests. Can/shall we vote for or against using the binary encoding?

I noticed another thing:

In the trajectory format the fill value for latitude and longitude is -99999, whereas in the profile format the fill value for latitude and longitude is 99999. Using the same value would be nice. Since profiles have to be reprocessed anyway I suggest -99999, even though all (?) other fill values are positive.

From : Bob Keeley

Date : 13/12/2002

As both Ann and Charles mentioned, if we generate multi-profile files registered to a single pressure array, we will have many missing values in individual profiles and large file sizes. My first concern is that the GDACs are doing the same thing. I understand that this is the case and that
is most important.

I believe that not registering the data to a single pressure array generates additional work for a user. Is there something we can do in the data system to help reduce this additional work?

From : Bob Keeley

Date : 15/01/2003

Subject : additional QC information

Concerning the updates proposed by Thierry and comments provided by others.

1. I agree with Annie's comments on wording.
2. I wonder why Claudia suggests 60 PSU as a max on salinity. In all of our archives we have never seen >41. I know there are places in the Red Sea near the bottom where salinities exceed this, but I understand they exceed 100!. If we want a nice max and min to bracket plotting of values then why not 0 and 50 or 0 and 45?
3. I question -2.5 as min for temperature. Water freezes at ~ -1.8 at typical oceanic salinities. A min of -2 brackets sensible values nicely for plotting.
4. I agree that fill values on lats and longs should be consistent between trajectory and profile files. I see no strong reason why -99999 is preferable to 99999 but don't much care. If we do not need to recreate trajectory files then lets make profile match trajectory.
5. Platform identification as a test that could be discarded in my view for the reason Claudia gives. We have yet to resolve how to record test failures. I received two responses from AST members to the question about recording test failures at each level; one in favour, one opposed. I think it is up to us to decide. We need to make the case for our choice. Once we decide this, let's then decide how to do the encoding.
6. I suggest that Thierry go ahead to make the changes in the Manual for

what we have agreed but to hold off distribution until items 2-5 above are settled.

From : Claudia Schmid

Date : 22/01/2003

Subject : additional QC information

> 2. I wonder why Claudia suggests 60 PSU as a max on salinity. In all of our
> archives we have never seen >41. I know there are places in the Red Sea
> near the bottom where salinities exceed this, but I understand they exceed
> 100!. If we want a nice max and min to bracket plotting of values then why
> not 0 and 50 or 0 and 45?

Partly because it was 60 before the upgrade. 50 is fine with me, it will probably include most salinities found in the Florida Bay (where we hope the floats won't go) and other similar regions. Any other votes?

> 3. I question -2.5 as min for temperature. Water freezes at ~ -1.8 at
> typical oceanic salinities. A min of -2 brackets sensible values nicely for
> plotting.

I suggested -2.5 because this is the value we are suppose to use in the global range test for temperature.

From : Claudia Schmid
Date : 24/01/2003
Subject : problem with DC_REFERENCE

I just noticed that we may exceed the permitted length for DC_REFERENCE (currently STRING16)

Our DC_REFERENCE looks like this:
"0196_WHOI26_005" which is the file name (without extension) of the original data file. This string is currently not longer than 15 characters, but it is likely to become longer. E.g. if WHOI deploys float 1000, or if they choose to use longer IDs for another reason we may exceed the length 16 allowed for DC_REFERENCE.

I suggest that we increase the permitted length for DC_REFERENCE to STRING32. What do you think?
TC : ok

From : Bob Keeley
Date : 24/01/2003
Subject : additional QC information

I have read through the latest version and have the following comments as I hit them. Most are minor. In some cases I have raised new questions that I would like othes to consider or address the questions you raised. These appear in points 4, 6, 7, 10, 11, 12, 17, 19, 23, 24, 27, 29, 30.

1. Section 2.2.2, Data_Type: The second sentence should end with ".. reference table 1."

2. Section 2.2.3, Platform_Number: The example has Q6900045. The Q should be removed.

3. Section 2.2.3, Data_State_Indicator: Rebecca, Gary and I are still sorting this out. Until otherwise told, what you have is applicable.

4. Section 2.2.3, Latitude and Longitude: I prefer a fill value of 999999 for these and used consistently everywhere lat and long appear.

5. Section 2.2.3, Positioning_System: In Positioning_System:conventions I would write it as = "ARGOS" or "GPS". I suggest the Example be : ARGOS not both as shown.

6. Section 2.2.3, Profile_Pres_QC: The comments refer the reader to the Handbook for an explanation of the flagging policy. I don't see that the handbook explains how we want the overall profile QC flags set. I suggest we replace the sentence stating "The flagging policy ...", for this and the next field, Profile_<param>_QC, to read "The overall flag is set to be the worst flag found at any level in the profile. From best to worst the flag order is 1,5,2,3,4,0"

7. Section 2.2.4. I would suggest adding a few more words to the introduction to this section to help clarify how to use the fields. I suggest the following "This section contains information on each level of each profile. Each variable in this section has a N_PROF (number of profiles), N_LEVELS (number of pressure levels) dimension. The original data received from the float and passed through real-time quality control should be placed in the <Param> and the QC flags set by the real-time process should be placed in the <Param>_QC field. Duplicates should be placed in the <Param>_Corrected and <Param>_Corrected_QC fields. As the data pass through scientific QC, the values and flags in these *_Corrected fields will be altered to reflect the results. The values and flags in the <Param> fields should never be altered."

8. Section 2.2.4, <Param>: There are font variations in the definition that should be cleaned up.

9. Section 2.2.5: The text here is similar to the text I suggested for section 2.2.4. I have no problem leaving this text in place.

10. Section 2.2.5, Calibration_Date: It just struck me that it makes some sense to have this date in the same form as JULD, for example. Is there a reason why this should not be so? Writing dates of all kinds in a consistent way makes a lot of sense to me.

11. Section 2.2.6, History_Date: This has the same comment as raised in point 10. If we agree to the change in 10 we should change things everywhere.

12. Section 2.2.6, how to write QCP$, QCF$: This is still undecided and needs decision. We have few enough tests that we could write this a a string of 1's and 0's rather than in hex form. Related to this, is the point I raised in an earlier email about recording test failures at each lelve. Opinions from the AST were from 2 only, one in favour and one opposed. We must decide. My vote is not to do this unless one of us actually plans to use such information.

13. Section 2.3.3, Platform_Number: Same comment as in point 2 above.

14. Section 2.3.3, Positioning_System, Same comment as in point 5 above.

15. Section 2.3.3, DC_Reference. I would like to remove this from here. See note 17.

16. Section 2.3.4, Latitude, Longitude: Same comment as in point 4 above.

17. Section 2.3.4: The problem we have is that there is no way to attach a unique identifier to each location in the trajectory file. We woud like to do this and so would suggest adding a field called DC_Reference in this structure and have it match the same characteristics as for profiles. I would alter the wording to say this is a unique reference to a surface drift location.

18. Section 2.3.6: Same comments as in points 11 and 12 above.

19. Section 2.3.7: I suggest that this section be eliminated. I suggest the sentence should read "When no measurements are available on a trajectory, the <Param> fields should be filled with fill values." I also suggest that this text be placed under the present text in section 2.3.4.

20. Section 2.4.3, Platform_Number: Same comment as in point 2 above.

21. Section 2.3.4, Trans_System: Same comment as for point 5 above.

22. Section 2.5.2, Platform_Number: Same comment as in point 2 above.

23. Section 3, Table 3.3, Max salinity value: I vote 42. The QC Manual posted at IFREMER uses 0-41 PSU for salinity range, but having an odd number for an upper limit makes plot axis awkward. A limit of 40 I think it too small so 42 is my choice.

24. Section 3, Table 3.3, Min Temperature value: The QC manual says -2.5. Though this is lower than necessary (-2.0 is sufficient) for consistency I can accept -2.5.

25. Section 3, Table 3.3: The C_Format column of PSAL looks cut off though I think the information is found on the next page. Perhaps this can be cleaned up.

26. Section 3, Table 3.4: I assume you will insert the new entry into alphabetical order to match the rest.

27. Section 3, Table 3.7: We may wish to have another entry to signify scientific QC had been performed. This would make it easy to find in the history record. As it is, a user would have to look at the History_Institution and figure if it was a scientific place or not, or go the Data_State_Indicator to see what it said. If people favour this idea, I suggest the code SQ = Scientific QC.

28. Section 3, Table 3.8: Three more entires have been added. They are 855 = NINJA, no conductivity sensor, 856 = NINJA, SBE conductivity sensor, 857 = NINJA, FSI conductivity sensor.

29. Section 3, Table 3.11: I would vote to eliminate the platform identification test. I agree with Claudia that unless data can be attached to an identifier we would not see it anyway.

30. Section 3, Table 3.11: I would like to add another test to this table. It will be the next sequence number and the test will be "Scientific QC carried out according to the Wong procedures".

31. Section 3, Table 3.13: This is not referenced anywhere I could see in the rest of the document.

TC: OK

From : Bob Keeley
Date : 24/01/2003
Subject : ocean definition

Dear All,

I attach a reply to the question that I posed to Howard. His definition meets the "simple test". I refer you to the issue he raises about floats near Panama. Will this also happen near Cape Horn, or Indonesia?

Regards, Bob

Strangely that came up in discussions with Robin yesterday. We actually did come to a conclusion but a very simple-minded one, 50?S. That line passes just south of New Zealand, but passes over shallow water, so all of NZ is N of the "Southern Ocean".

On a similar topic, I note that the Argo Global Data Servers appear to define oceans based on allocating individual Marsden squares to one ocean or another. At least, I think that is how it is done. There is one Marsden square in the Panama canal area that actually includes

```
contributions from both Atlantic and Pacific Oceans.  I think it has been
designated a Pacific square.  I wonder what would happen if a float was
launched in the tropical Atlantic, passed through this square and appeared
for a short time to be a Pacific float and then became an Atlantic floats
again.  Would different profiles appear in different parts of the geo
directories?
Howard
-----Original Message-----
From: Bob Keeley [mailto:keeley@meds-sdmm.dfo-mpo.gc.ca]
Sent: Friday, January 24, 2003 10:52 AM
To:   'Freeland, Howard'
Subject:    Ocean definitions


Dear Howard,
As an official member of the AST, I would like your advice about defining
oceans. The question has come up about where the Antarctic ocean begins
and
the other oceans stop. Of course a nice simple minded definition would be
good, such as such and such a latitude, rather than something based on the
average position of the Antarctic convergence or whatever. Any wisdom to
offer?

-----Original Message-----
From: Thierry Carval [SMTP:Thierry.Carval@ifremer.fr]
Sent: January 24, 2003 10:51 AM
To:   Rebecca McCreadie
Cc:   Sylvie Pouliquen; Loic Petit de la Villeon;
mark.ignaszewski@fnmoc.navy.mil; Bob Keeley
Subject:    RE: Definition of Southern Ocean

Dear Rebecca,

You raised an important issue : argo data management has to use a common
definition of oceans.
In Ifremer, the ocean calculation of the 3 oceans is based on Marsden
Squares.
We do not handle the Antartic ocean for the moment.
I noticed that our ocean definition is slightly different from the NODC
(and
MEDS ?) definition.
For example, the Arafura sea is located in Indian ocean by Ifremer,
meanwhile it is located in Pacific for NODC.

Is it possible for Bob to issue a proposal for ocean definition within
Argo
project ?
This proposal will explain the calculation of the ocean from a given
location.
Best regards,

Thierry



> -----Message d'origine-----
> De : Rebecca McCreadie [mailto:rebl@bodc.ac.uk]
> Envoye : jeudi 19 decembre 2002 14:58
> A : mark.ignaszewski@fnmoc.navy.mil; Thierry.Carval@ifremer.fr;
> belbeoch@jcommops.org
> Objet : Definition of Southern Ocean
>
```

>
> Mark/Thierry,
> At the data management meeting it was stated that a new "geo" directory would be added for antarctic floats. As BODC is the Southern Ocean Regional data centre I was wanting make sure that we all use the same definition for the northern extent. This would then be the southern extent for all the other oceans. At the moment the only suggestion made to me was 30pS but was not based on any physical reason which I'm aware of. I feel is rather a long way north. What were you planning to use?
> >From Rebecca
>

TC : the ocean definition is now in the user's manual (28/04/2003)

From : Annie Wong
Date : 27/01/2003
Subject : additional QC information

I have read version 1.0a and Bob's comments. The following are in reply to Bob's points:

4). No preference on fill value for lat and long.

5). Please use a consistent format (ARGOS or Argos) for positioning system (p.14), positioning accuracy (p.24, in columns 2 and 3), and Table 5 (p.44, in title and heading of table).
TC : OK

6). Agree with Bob's suggestion.

7). Agree with Bob's suggestion, but suggest slight re-wording in ..
"The original data received from the float and [EXAMINED BY] real-time .."
I think this clarifies that the result of the real-time qc process is to provide flags, but it does not alter the values.
TC : OK

10). No preference on what format dates take. But please correct on p.17, colume 2, should be CALIBRATION_DATE instead of DATE_TIME.
TC : OK

12). I have no plans to use details of test failures at each level.

I have no comment on the rest of Bob's points. The following are a few minor things I picked up:

a). p.22, Section 2.3.3, for TRAJECTORY_PARAMETERS, please correct
"The parameter names are in the GF3 code list".
TC : ?

b). p.25, Section 2.3.4, "PRES_CORRECTED values come[] from PRES ...
If no calibration is available or if no other corrections are made, PRES_CORRECTED contains the same values [AS] PRES."
TC : ?

c). p.35, Section 2.4.5, for SENSOR, please correct
"The parameter names are in the GF3 code list".
TC : ?

From : Claudia Schmid
Date : 27/01/2003
Subject : additional QC information

```
p. 14, <PARAM>CORRECTED
      add reference to Table as in <PARAM>
```

DC_REFERENCE in trajectory files: I don't think it is necessary to have a DC_REFERENCE associated with each surface location (Bob's point 17). I think the cycle numbers and times together with an overall DC_REFERENCE are sufficient.

Table 3, Conducitivity: The formats should read %10.4f, F10.4, 0.0001f (as I explained earlier)

Table 3, Oxygen: I suggest %9.3f, F9.3, 0.001f

Positioning system: I think ORBCOMM can also estimate float positions.
I was informed by WHOI that they sometimes have position fixes for ORBCOMM floats that do not come from GPS on the float. You might want to ask them about more details?

Ocean codes:
How are the oceans defined?
What about floats that go from one ocean to the next?
Where is this information supposed to be stored?
TC : this information is stored in the GDAC index. There is no ocean code in the Argo NetCDF files.

Bob's point 12 (QCP$, QCF$): I favor the use of strings of 1's and 0's.

From : Yeun-Ho Chong
Date : 04/02/2003
Subject : a Fill_Value for NetCDF variable of NF_CHAR

We would like to share what we found out about the Fill_Value for a character variable.
The latest user's manual suggested that we use '00000000000000' for the CALIBRATION_DATE FillValue in writing profile and '99999' for the TRANS_FREQUENCY in writing meta file.

You will find it from the following e-mail to Steve Emmerson of unidata.ucar.edu, only a single character is allowed for a fill_value for a character varaible in NetCDF.

TC : Thank you for your remarks. So here is the latest definition of CALIBRATION_DATE and TRANS_FREQUENCY :

```
Char CALIBRATION_DATE (N_PROF N_CALIB, , N_PARAM, DATE_TIME)
        CALIBRATION_DATE:_FillValue = "0";

char TRANS_FREQUENCY(STRING16);
TRANS_FREQUENCY:long_name = "The frequency of transmission from the float";
TRANS_ FREQUENCY:units = "hertz";
TRANS_ FREQUENCY:_FillValue = “9”;
```

From : Bob Keeley
Date : 26/02/2003
Subject : trajectory files

I received an inquiry regarding two trajectory files. The ones in question were 2900193_traj.nc and 6900199_traj.nc. The first is a file we create here. In it, I note that we do not preserve the order of variables as described in the latest version of the manual. I expect this is not serious for people who are using the netCDF libraries to read the data, but
if you think it is better, we will adjust the files so the order is identical to that in the manual.

The second file uses a different name for trajectory_parameters (at least this one difference). I looked in an older version of the formats manual and sure enough what is used is in version 1.2b from 2001. I believe
this is a UK float but there is nothing in the data_centre field to identify who did the processing. Do you know who does this? We should notify them that they are using an older version.

TC : Here are 2 answers to your 2 questions :

1. Do we have to preserve the order of variables :
This can be a problem for people that do not use netcdf libraries to read data, but transform files into text before manipulating the data.
But, I am not in favor of saying that the order of varaibles is fixed in the
format. For instance, Claudia recently inserted additionnal variables in her
files, that are not for the moment mandatory. In the near future, corrected
values (or correction on values) will become optionnal (they will be present
if there is a correction available). Therefore, it will not be possible to have a strict order of variables.

2. Trajectory file 6900199_traj.nc :
You make me blush because this file was produced by Coriolis. Until february, we did not fill correctly the data centre variable.
All Coriolis trajectory files have now been re-processed and comply with format 2.0 . I append to this mail the latest update of 6900199 trajectory file.

From : Claudia Schmid
Date : 27/02/2003
Subject : directory files

Essentially, I agree with Bob and Charles.

I'm not sure what the advantage of the XML tags is (maybe you can tell me). This depends on what these files are supposed to be used for. For me the XML tags are only a part of the file that I will have to skip over when using the files.

I would use the directory files for an automatic ftp transfer by eliminating all information that is not a file or directory name.

This would be unecessary if one of the following two formats is used:

format (1) A format that can be easily converted to the format (2) or another format, e.g.

```
/root_dir/jma/29051/profiles R13857_001.nc
/root_dir/jma/29051/profiles R13857_002.nc
...
/root_dir/jma/29051/profiles R29051_002.nc
```

This format can be space, comma (or ...) seperated.

format (2) A file that can be directly used for the ftp session, e.g.

```
cd /root_dir/aoml/13857/profiles
mget R*.nc
...
cd /root_dir/jma/29051/profiles
mget R*.nc
bye
```

This format is quite limited and not very convenient to look at, but it is nice for users who download all data using ftp.

I'm not sure how important the other information in the directory files is, but it is possible to add it to format (2) whereas there is no room for it in format (1).

If additional information is added to format (2) it is essentially the format as you described it, but it spares users that do not use XML the removal of the header and trailer with the XML tags.

TC : My previous mail sent to argo-dm-format mailing list describes the advantage I see in using xml : we can add additional informations with a standard syntax in the beginning of the file.

Your simplified format proposal is correct for someone who wants to download all argo files.
But, the additional columns of the proposal permit to select (or perform statistics) on the following criteria :
date,latitude,longitude,ocean,profiler_type,institution,date_update

From : Charles Sun
Date : 27/02/2003
Subject : gdac directory files

There should be some reasons that people start using XML for data Sharing and exchanging. It is not a bad idea to keep XML files online but I would also like to see a CSV version of the files online too, if this is not a burden of the GDACs.

From : Thierry Carval
Date : 10/03/2003
Subject : Ocean definition

Following the 5th IAST meeting decision, here is a proposal for ocean definition within the Argo project.

A : Atlantic ocean area
I : Indian ocean area
P : Pacific ocean area
N : Antartic ocean area

Antartic ocean is below 40°S.
The indian/pacific frontier is 145°E.
The Atlantic/Indian frontier is 20°E.

From : Bob Keeley
Date : 12/03/2003
Subject : delayed mode data questions

We are in the final stages of getting our delayed mode data ready to go to GDACs. I have some questions I would like your advice on.

First, we are requiring our PI to write additions to QCP$, QCF$. The think we should add a test number for delayed QC to indicate the Wong et al
procedure was performed (or failed) and also one for visual QC by the PI. We can use the next 2 larger 2**n in table 3.11 or try to reuse 64 and 1024
which are not used. I see no harm in using 2**16 and 2**17 if you agree. The question is, do you think we should add these two entries to table 3.11
and if so what numbers should we use?

Second, you may have heard from Sylvie that using the word "CORRECTED" causes some concerns. However, in talking to Dean his solution was to call it "ADJUSTED" instead but to continue to use these fields as we described in the 24 Jan 2003 version of the netCDF Users Manual. Having said that, my
question is do you feel we can move forward to annouce this version of the format and should we start to send data to GDACs in this version or the present form without the CORRECTED_ERROR fields? For the delayed mode data we have in hand, no corrections/adjustments were necessary.

As an additional question, but one not needing an answer immediately, we
need to have a strategy for controlling the implementation of format changes. Your thoughts on this would be welcome.

TC : 23/04/2003
Here is a version 1.0b of the users's manual.
This is a draft that will become a version 2.0 as soon as it is validated.

I added 2 new entries in table 3.11 :

131072	Wong et al correction
262144	Visual QC performed by PI

I also replaced CORRECTED with ADJUSTED in the draft of the user's manual.
In the proposal, <PARAM>_ADJUSTED, <PARAM>_ADJUSTED_QC values are not mandatory.
They appear in the file is an adjustment was performed.

The strategy for controlling the implementation of the new formats will be detailed later.
We first have to agree on the new format. My feeling is that the GDAC will accept both formats during a certain time.
The format checker of the GDAC will read the format of the file (2.0 or 2.1)
and will then apply the corresponding checks.

From : Claudia Schmid

Date : 11/04/2003
Subject : file naming conventions

> A cycle may have 2 profiles (descending and ascending). This was a
> "standard" feature of the first provor.
> According to Argo users's manual, the convention is to append a 'D' to the
> cycle number of the descending profile.

Thank you!

FYI: I searched for this information in the user manual and did not find it (in the January and the March version). Now I found it in the GDAC document (argo_gdac_servers.pdf). I recommend adding the file naming conventions to the user manual (seperated by the four data types, at the beginning of each of the sections for the four different formats).

TC : The file naming convention is now described in the data access chapter (§4)

From : Steve Loch
Date : 17/04/2003
Subject : full values for character variables

For some reason the fill value for character variables has not been defined. The result is that some people pad with binary zeroes and others with blanks. As there is supposed to be compatibility with Fortran I would think blanks are to be preferred. Conversely with C (and Java, etc.) you can completely fill the dimension in which case there is no binary zero to terminate the string. Again standardising on a _FillValue of blank would indicate exactly what to expect.

There is incidentally a bug in the NetCDF C API in a relation to a somewhat similar problem with string attributes (reported but doesn't appear on the UCAR Web site when I last looked).

From : Steve Loch
Date : 29/04/2003
Subject :

These comments relate to User's Manual 1.0b. Apologies for not commenting earlier on these issues but we are only now getting round to actually using the formats as specified: as I understand it not a lot of data has accumulated in these formats, particularly in the areas referred to, and so amendment may still be possible (absolutely necessary in the first case).

We'll start with the most obvious item.

(1) On pages 21 and 22 there are references to DC_REFERENCE in the same (trajectory) file albeit with different dimensions. This is not allowed in NetCDF. The reference on page 22 is newly installed and refers to 'location'. Who needs this?
TC : DC_REFERENCE is removed from page 21. MEDS (Bob Keeley) wants a DC_REFERENCE for each measurement.

(2) There is a mismatch between Profile and Trajectory Formats in relation to time and position.

The Profile format clearly recognises that the position and time of surfacing can be different from the position and time of transmission.

In the Trajectory file measurements precede, possibly by some hours, the transmissions from whence positions are determined but no allowance is made for the fact. Why is this? Is the surface reading of the profile also to be included within the Trajectory file?

(3) A cluster of issues relates to history in both Profile and Trajectory files.
In the Profile case there is no unchanging identifier for the item being edited.

We have HISTORY_START_PRES and HISTORY_STOP_PRES which can be used to bracket a section of a profile but could be used to identify a particular measurement - by for example setting them equal to the chosen sample's pressure or close enough to exclude other samples. Only if we adopt the latter approach do we identify the value concerned but then this seems to be a clumsy way of doing things.What happens if the pressure is itself adjusted as part of the editing process? Do all the codes in reference table 7 alter just the value or just the flag? If they do both how do we store the previous values. What is the update program (UP)? What for example is the difference between CV and ED.

None of this would matter much except for the fact we apparently have to record the previous value thereby limiting any edit description to a single-value or single-flag edit. Do we really need to do this? Probably not as there are contra-indications alluded to above - but I'm guessing. In other words it would be acceptable to have a fill value for multi-value edits - but this issue needs to be made plain.

(4) If we do want to record the previous values for all actions then steps should be taken to go to, what in relational terms, is normal form. I.e. information about the edit step is recorded along an edit dimension (which would not be unlimited but could be set up initially with, say, 20 locations and have the previous values inhabit the unlimited dimension cross-referenced through edit ordinal (1,2,3, etc.). Only after the initial 20 edits would the file require rebuilding. Obviously the unused space would be initialised to null.

I think when we were undertaking the review that we agreed, subliminally anyway, to avoid the issue of normal form but if we ever adopted that approach there are other candidates. E.g the Technical Information file. The problem is that as the file extends through the course of time you may get small unintended deviations in parameter naming. Might it not be better (and space saving) to identify the parameter once and then refer to it by ordinal?

From : Bob Keeley
Date : 29/04/2003
Subject :

We also had noticed the use of DC_REFERENCE in two places. The one we wanted was the one on page 22 since this allows us to provide a unique identifier to each trajectory location. We can live with the 2 but this might be confusing. If we have to live with one, the dimension must be string32.
TC : we keep only one DC_REFERENCE. It is defined for each location (chapter 2.3.4 in page 22), its dimension is string32.

The surface reading of the profile should also be in the trajectory file. If there is no position, we include the time and put fill values in the lat and long fields. This will permit a user to interpolate if they choose.

Start and Stop _Pres are a convenient way in some cases to record what has happened over a range of pressures, but not so nice when something has changed at a single pressure. The start and stop values are those as appear in the adjusted profile after the action is taken. The variable affected, including changes to pressure, appear in Parameter. The previous value records what was present before the action was taken. If a variable is adjusted by some amount and this is done consistently over more than one pressure, we still need to record the actions at the individual pressures in separate history records because that is the only way to record previous value. If however, a whole suite of values are flagged something and we wish to reset all of them to something else, a single history record can do this since the previous value will hold the flag (that is the same at all pressures in the range).

The codes in table 7 affect different things. CV is used for changing a value and CF for changing a flag. CV and ED would seem to be redundant. You can also use the history records to document the processing stages through which the data passed. We do this at MEDS recording when data pass through a reformat, QC, update, etc. If these programmes happen to alter the contents of the file, we record these changes as well. We have found that recording the processes helps to unravel problems at a later date. Regarding using "normal" forms, you have a point to be considered. I do not wish to make such changes now. The history part is awkward because it appears in conjunction with a raw and an adjusted profile. In some sense, the original reason for having it has changed. This is also an area for future consideration. We will be having a discussion at the upcoming meeting in Monterey concerning format changes and change control. We can consider these issues then.
TC : OK

From : Steve Loch
Date : 30/04/2003
Subject : character fill value

The safest option is to define a fill value of blank for character variables. I have had some experience of problems with Fortran where binary zeroes are included in character data - with some compilers you lose the rest of the data. In the context of padding (on the right!) this doesn't matter - because of course there's no further information of interest to lose. So far as I'm aware this is the only "problem" but there could be others which is why I say this is the safest option.

One problem of blanks though is that the ncdump facility - which many people seem happy to use - will print out all the blanks whereas binary zeroes get suppressed. With the amount of padding in these files this might be a major downside to listing with ncdump.

There seem to be four options.

1) Choose blank. State this in the preamble to the format definitions.
2) Choose blank. Put the fill value in explicitly as an attribute
3) Choose binary zero. State this in the preamble to the format defintions
    Check out a representative set of Fortran compilers for potential problems.
4) Choose binary zero. Put fill value in explicitly as an attribte.
    Check out a representative set of Fortran compilers for potential problems.

(1) has the advantage of minimal change to the document but (2) is more realistic and makes use of NetCDF facilities. (4) (and (3)) is obviously more time consuming but might be the best option if we have the resources to spare for the assessment exercise. I haven't checked out yet how CDL handles binary zeroes in the context of character data - judging by the printout of ncdump, not very well!

TC : OK for solution 2.

From : Thierry Carval
Date : dates
Subject : 30/04/2003

I would like to have your opinion on the date format with Argo Netcdf. Measurement dates are stored in Julian Date (JULD). This format is fine for programming, but not easily readable by users.

We decided that the format version 1.0 would not handle a string date (YYYYMMDD HHMISS) for measurements, because we should avoid redundant information (what to do if the julian date and the string date are different ?)

Do we change our opinion on that : do we add a string date to the Julian Date ?

Example :

```
double JULD(N_PROF);
JULD:long_name = "Julian day (UTC) of the station relative to REFERENCE_DATE_TIME";
JULD:units = "days since 1950-01-01 00:00:00 UTC";
JULD:conventions = "Relative julian days with decimal part (as parts of day)";
JULD:_FillValue = 999999.;

char DATE(DATE_TIME);
DATE:long_name = "Date of the station";
DATE:conventions = "YYYYMMDDHHMISS";
DATE:_FillValue = " ";
```

BK : Why should we reconsider this? The same concern still applies. I think we should leave this unchanged.

6. Comments on user's manual version 1.0c

From : Thierry Carval
Date : 30/04/2003
Subject :

Here is a version 1.0c of the Argo users's manual.
This is a draft that will become a version 2.0 as soon as it is validated.
The differences with version 1.0 are typed with a green font.

The main issues of this new version are :
1. How to handle delayed mode data : read §2.2.4 and §2.3.4
2. Definition of GDAC index files : read §2.6

From : Claudia Schmid
Date : 30/04/2003
Subject :

thanks for your work. Here are some comments (you can keep them on the pile of sugestions for the next version):

An subsurface drift (eg : 10 days),
--> A subsurface drift (eg : 10 days),
TC : corrected.

Profile format:
Add file name conventions or refer to section 4.1
TC : done.

DC_REFERENCE(N_PROF, STRING3216); (STRING3216 problem was detected before)
TC : we use string32 as was asked on january 24th 2003. Do we need more ?

Trajectory file format
Add file name conventions or refer to section 4.1
TC : done.

When no measurements are available on a trajectory, the
temperature parameter is set with fill values.
--> When no measurements are available on a trajectory, the
parameter is set with fill values.
TC : I think that we need at least one parameter because the dimension N_PARAM should not be set to 0 (which means unlimited dimension for NetCDF). So the convention is to set N_PARAM to 1, to declare a temperature parameter and to set this parameter with fill values.

2.3.7 Trajectories with no measurements
section has no content
TC : this section has been removed.

Meta-data format:
Add file name conventions or refer to section 4.1
TC : done.

Technical information format
Add file name conventions or refer to section 4.1
TC : done.

Reference table 3 : parameter code table
CNDC - we need and use (I know that you may already have this
listed for change in a later version):
CNDC:valid_max = 11.f ;

```
                CNDC:C_format = "%10.4f" ;
                CNDC:FORTRAN_format = "F10.4" ;
                CNDC:resolution = 0.0001f ;
```

TC : done.

Reference table 11 : QC Test Ids
  is not mentioned anywhere as far as I can tell.
TC : HISTORY_QCTEST description for profiles and trajectories now refers to table 11.

From : Mathieu Belbeoch
Date : 02/05/2003
Subject :

In order to have a complete "Argo Data Management User's Manual", I would suggest to add a section about the GTS data publication, and the description of GTS data formats (TESAC and BUFR) and to update the section 1.1.

JCOMMOPS offers its participation for the writing.
TC : to mention GTS formats (TESAC and BUFR) in Argo user's manual, I added an additional chapter 4.2 : Other data sources.

Misc. details:
- to change the Argo logo with the new one
- 3000 floats target: 2006 not 2005 ...
- table 8: add code 858 for NINJA with TSK conductivity sensor

TC : done.

From : Bob Keeley
Date : 02/05/2003
Subject :

I disagree with Mathieu. If we add the description of the TESAC and BATHY code forms, then we have to reproduce what is done elsewhere and our version has to keep this up to date, including code tables. I also don't think the User's Manual is the place for details about the GTS. This is an existing system, operating independent of Argo. Let's just point people to more information. If we want to provide greater details, I would put that in the Data Management Handbook because that is where the entire system is described.
TC : to mention GTS formats (TESAC and BUFR) in Argo user's manual, I added an additional chapter 4.2 : Other data sources.

From : Mathieu Belbeoch
Date : 02/05/2003
Subject :

I agree with you on the point that this is not the place for providing details on code tables, etc.
A few links pointing at the right place will be appropriate.

Whatever, it is the place to put information on the BUFR template used by Argo , that will be strongly linked to netCDF format.
TC : to mention GTS formats (TESAC and BUFR) in Argo user's manual, I added an additional chapter 4.2 : Other data sources.

From : Jon Turton
Date : 02/05/2003

Subject :

If I can put my "oar" in here. While I agree that GTS operates independently of Argo we must recognise that many Argo data users do (and will continue to) rely on the GTS for the receipt of Argo data. Therefore the GTS must remain an integral part of the Argo data system.

However, there is no point in repeating information that is compiled and maintained elsewhere in the Argo Data Manual - but we should provide references or links as appropriate.

TC : to mention GTS formats (TESAC and BUFR) in Argo user's manual, I added an additional chapter 4.2 : Other data sources.

From : Bob Keeley
Date : 02/05/2003
Subject :

I am not sure what is the best strategy for the BUFR code. This will be tailored to Argo and so we will need something that goes into appropriate detail of our coding, with suitable links to more general BUFR documentation. At this moment, and not having really thought about this, I kind of think we might want a separate document. However, another section to the Users Manual may be more appropriate. I would need to think more about this.

TC : to mention GTS formats (TESAC and BUFR) in Argo user's manual, I added an additional chapter 4.2 : Other data sources.

From : Takashi Yoshida
Date : 04/05/2003
Subject :

I would like to make a comment although I did not see the version 1.0c because probably the e-mail you sent was larger than 2MB and it was not accepted our e-mail system.

TC : you are not registred in argo-dm-format@ifremer.fr mailing list. I will register your email address.

I guess that the table 8 does not contains code 858 but contains 855-857, based on the Mathieu's comment. It's strange that only 858 is not described in the table because those four codes are now in the same status, that is so called pre-operational. So, I ask you to add the code 858 to the table as Mathieu asked.

TC : done.

Regarding to the additional section about the GTS, I agree with Jon. Appropreate references or links seem to be enough.

TC : ok.

From : Mathieu Belbeoch
Date : 05/05/2003
Subject : BUFR template

I don't want to see a complete description of the BUFR code in the manual. However, the BUFR template for Argo is something that will develop and change (as netCDF).

The details have to be discussed inside the Argo data management group and the description of the template have to be in the document.

I recall that the first BUFR test-messages are being decoded via the Argos GTS chain:

Asked by DBCP, Argos is upgrading its system to be able to deal with BUFR and Etienne Charpentier is testing it for Argo float data too.

TC : ok, that test is interesting.

From : Ann Thresher
Date : 06/05/2003
Subject :

Only one question so far - in the new trajectory format, the field TRAJECTORY_PARAMETERS has a fill value available - so presumably if we don't measure parameters along the trajectory, we can declare N_PARAM=1, set the TRAJECTORY_PARAMETERS(1,:)=' ', and then don't define or report any <PARAM> variables (they become optional).

But the manual then states that, if no measurements are available on a trajectory, the TEMPERATURE parameter is set with fill values. As I read this, there should be no temperature parameter declared. Or should we set the TRAJECTORY_PARAMETERS(1,:)='TEMP' even though this is not actually true, and then have a TEMP variable field with blanks? Which then also requires the TEMP_QC, TEMP_ADJUSTED and TEMP_ADJUSTED_QC fields. Seems pretty silly, actually.

The real problem here becomes, if someone wants to find "all trajectories with temperatures measured", they end up with all the trajectories, regardless, which really limits the usefulness of this field in particular.

I prefer the first process. Guidance?
TC : Your remark on trajectory measurements is correct.
It seems strange to declare a temperature parameter when no parameter is available.
But, we cannot set N_PARAM (number of parameters) to 0 because this value means unlimited dimension.
So we set N_PARAM to 1 and declare a temperature parameter, all temperature values are then set to fill values.

An other (probably better) option to handle trajectories with no parameters measured is to set N_PARAM to 0 and to remove any field with a N_PARAM dimension : PARAM, PARAM_QC, PARAM_ADJUSTED, PARAM_ADJUSTED_QC and TRAJECTORY_PARAMETERS.
This option may complicate the softwares handling profile files (you have to test the value of N_PARAM and in that case perform differently). The first of these software is the format file checker running on GDACs.

From : Ahn Tran
Date : 13/05/2003
Subject :

I found a small mistake on the User's manual Version 1.0c. In the manual, it said that the Fill Value for Handbook_Version is a blank, but Handbook_version is a float number. HandBook_Version:_FillValue = " " occurred in profile, trajectory, meta and technical netcdf file.
TC : I propose to declare HANDBOOK_VERSION as FORMAT_VERSION is declared : STRING4, with a blank fill value.

And I am also wondering when will you accept this format at the GDAC ?
TC : Mark Ignaszewski is preparing a document on the format change implementation.

From : Ann Thresher
Date : 13/05/2003
Subject :

Looking in detail at version 1.0c, I find a problem with the fill values in the trajectory history section - you cannot have a float fill value in

a variable that is defined as 4 characters long - particularly when that float value is 5 characters long. Should this be " " or ""? And should the date fill value be the same as the fill values used elsewhere for dates, i.e., ""?
TC : yes, the fill values of trajectory history section are not correct. The fill values have been changed to " " (blank), they are identical to profile history section.

From : Annie Wong
Date : 13/05/2003
Subject :

I have two suggestions:

1). Section 2.3.4, if delayed-mode data are to appear in Trajectory files, then the parameter <PARAM>_ADJUSTED_ERROR should be included;
TC : ok, I added <PARAM>_ADJUSTED_ERROR.

2). Reference Table 13: Ocean Codes ... why not use Southern Ocean (S), instead of Antarctic Ocean (N)?
TC : I agree with that suggestion, we should adopt it if there is no one against it. But, the interest of a Southern ocean is still not clear among scientists. Recent mails from Howard Freeland, Brian King or Susan Wijffels are against the handling of a 4th ocean. They seem to prefer 3 oceans only (A, P, I).

From : Lin Shaohua
Date : 14/05/2003
Subject :

I have a suggestion.
In my opinion, at the beginning of Argo project, Argo data manual should be a convenient tool for all users. So the GTS code publication should be added to the data manual. In the future when users are familiar with Argo data, we can get rid of them and provide appropriate references or links.
TC : to mention GTS formats (TESAC and BUFR) in Argo user's manual, I added an additional chapter 4.2 : Other data sources.

From : Ann Thresher
Date : 19/05/2003
Subject :

I just noticed something in the new user's manual - the HANDBOOK_VERSION (all data files) fill value is ' ' or blank - I don't think you can have a character fill value for a float variable....

I suggest 0 or -99 or see if there is a standard we should use?
TC : I propose to declare HANDBOOK_VERSION as FORMAT_VERSION is declared : STRING4, with a blank fill value.

From : Ann Thresher
Date : 23/05/2003
Subject :

I think the best solution is to make the <PARAM> fields (and all the CORRECTED and QC fields, etc) optional, set the N_PARAM to 1 (0 doesn't really work...) and then use it only in the TRAJECTORY_PARAMETERS(N_PARAM,STRING4) variable where we use the fill value to indicate that there are none. This solves all problems with anyone who wants to select trajectories with TEMP fields because there won't be any... Or - we can declare all the TEMP associated variables and fill them with fill values while still setting the TRAJECTORY_PARAMETERS(N_PARAM,STRING4)to the fill (null) value. Then, if anyone wanted TEMP parameters associated with the trajectories, they could check the TRAJECTORY_PARAMETERS variable. This would have no TEMP entry so they could skip that trajectory file. Either way, it seems to make most sense to use the fill value for the TRAJECTORY_PARAMETERS if no parameters were measured. Otherwise, it is just misleading....

Yes? No? Which? Thanks!

TC : here is a proposal to handle trajectories with no parameters measured is to set N_PARAM to 0 and to remove any field with a N_PARAM dimension : PARAM, PARAM_QC, PARAM_ADJUSTED, PARAM_ADJUSTED_QC and TRAJECTORY_PARAMETERS.

From : Takashi Yoshida
Date : 23/05/2003
Subject :

I would like to make comments on the draft.

1) The length of PROJECT_NAME is STRING32 in metadata format, but STRING64 in profile and trajectory formats. There is no reason to keep inconsistency between those formats for the same variable. I have no idea on the choice out of the two. But it should be the same length.

TC : PROJECT_NAME is now STRING64 for all formats.

2) How do you think about introducing first and last times of transmission received to the trajectory format? They both come from the time attached to the ARGOS messages, not from any estimation, and so it seems to be worth to keep them in order to trace the movement of floats at surface. In some cases, such as major clock drift for APEX, they are useful to make better estimation of JULD_DESCENT_START and JULD_ASCENT_START.

TC : see the answer from Claudia Schmid on 03/06/2003.

From : Bob Keeley
Date : 23/05/2003
Subject :

I have only one comment and that concerns the ocean codes of table 3.13. Brian King suggests we make an executive decision to delay creating a southern ocean directory at the GDACs. I am comfortable with this. So, the comment you made to Annie is correct. What I don't understand is what relevance the ocean codes have at all. Version 1.0c does not reference this table, and in looking at the IFREMER ftp site I did not see where these codes are of use. It seems to me that this section (3.13) that describes the oceans belongs in the data management handbook where the discussion talks about how the ftp site at the GDACs is organized. It doesn't belong in this manual at all. If we put this in the Handbook, we do not need the table at all, simply the figure and the text below it. Following Brian's suggestion remove all reference to the southern ocean and the line at 40S.

TC : The ocean codes are used in the GDAC FTP directory file format (§2.6). The description of each profile file includes the ocean code. That is the reason why I think that we need a table of ocean codes.

I will remove the Southern ocean from the table. Do you agree with that ?

From : Claudia Schmid
Date : 03/06/2003
Subject :

> 2. First and last time of transmission for each cycle :
> I suggest that we declare the 2 following variables to the trajectory format
> (in cycle information section) :
> JULD_FIRST_RECEIVE : Julian day (UTC) of the first data reception by the
> telecommunication system.
> JULD_LAST_RECEIVE : Julian day (UTC) of the last data reception by the
> telecommunication system.

We currently store this information in the following way (and I suggest to adopt this as the official way):

we have the first and last transmissions in JULD (, ...) with fill values in the position and drift data.

```
E.g.:
JULD     = 1,2,3,4,5,6,7,8,9
LATITUDE = 1,_,_,_,5,6,7,_,_
PRES     = _,_,3,_,_,_,_,_,9
```

JULD 1 (7) and 5 correspond to the last and first position.
JULD 2 (8) and 4 correspond to the last and first transmission.
JULD 3 and 9 correspond to the drift phase.

I think having the first and last transmission as part of JULD makes sense. Then everything is in the same `place' and it is not necessary to introduce additional variables to store this information.
TC : I agree with Claudia, the date and time of the JULD in the trajectory format are realated to the location received from the positionning system (ex : Argos). For each cycle, a series of locations is received and stored without estmation or interpolation.

From : Takashi Yoshida
Date : 05/06/2003
Subject :

I am thinking about the possibility that someone wants to store any event time in JULD with fill values in LATITUDE, LONGITUDE, POSITION_*, and <PARAM>*. In that case, the first and last time of transmission received might not be distinguished from other event time. Introduction of the additional two variables seems to be better because we don't have to care about this kind of confusion.

However, I cannot think of any example of such a case at the moment. If no one think of such a case and we don't have to suppose such a case, then I will follow Claudia's suggestion and will suggest to add a sentence which tells that JULD with fillvalues in LATITUDE, LONGITUDE, POSITION_*, and <PARAM>* is Julian day of the surface transmission received and that those of the first and the last transmission received should be stored in JULD.
~~TC : if we consider that it is important to store the first and last time of transmission for a cycle, I suggest that we declare the 2 following variables to the trajectory format (in cycle information section) :~~
~~JULD_START_RECEIVE : Julian day (UTC) of the first data reception by the telecommunication system.~~
~~JULD_LAST_RECEIVE : Julian day (UTC) of the last data reception by the telecommunication system.~~
TC : see Bob Keeley's comment 10/06/2003.

From : Rebecca McCreadie
Date : 04/06/2003
Subject :

There is no fill value defined for trans_repetition (section float characteristics in metadata format). I assume there should be.

TC : Ok :

TRANS_REPETITION:_FillValue = 99999.f;

From : Ann Thresher
Date : 06/06/2003
Subject :

I always prefer to be explicit and so prefer defined variables for whatever should be stored. Why create confusion and require people to refer to a manual if we don't have to?

~~TC : Ann refers to Takashi Yoshida (05/06/2003) and Claudia Schmid (03/06/2003) remarks on start receive and and last receive time. I agree with Ann that these informations, if needed should be explicitly declared in the cycle information section (see comment 05/06/2003) .~~

TC : see Bob Keeley's comment 10/06/2003.

From : Rebecca McCready
Date : 04/06/2003
Subject : meta-data

I'm currently working on writing metadata files and have come across something that so far looks like everyone whos submitted files have done something different. In the section "float cycle information" there are 5 times; CYCLE_TIME, PARKING_TIME, DESCENDING_PROFILING_TIME, ASCENDING_PROFILING_TIME and SURFACE_TIME. With Apex you know the down time (PARKING_TIME + DESCENDING_PROFILING_TIME), and the up time (ASCENDING_PROFILING_TIME + SURFACE_TIME). With these floats it is only possible to get ASCENDING_PROFILING_TIME and SURFACE_TIME and it could be quite a few profiles into the floats lifetime to get these accuratly if you miss the test message at the time of deployment. It is not possible (as far as I or Brian King are aware) to resolve the down time into PARKING_TIME and DESCENDING_PROFILING_TIME. All you can do is make an assumption that the float takes 6 hrs (or some other amount of time as advised by Webb) for the DESCENDING_PROFILING_TIME.
Looking at what others have done it seems that there is no standard of what to put in these fields. Also, in the case of the file looked at from MEDS the 4 times (parking, descending, ascending and surface) didn't add up to equal the CYCLE_TIME.
The only sensible suggestion I have is for there to be a status field for each one so it can be indicated as to whether it is known or estimated (similar to the ones in the trajectory files).
Having spoken to Brian we have decided to put 0 in DESCENDING_PROFILE_TIME and put the down time in PARKING_TIME. This appears to be similar to what MEDS have done.

TC : ok, this is a content issue, not directly related to data format.

From : Bob Keeley
Date : 10/06/2003
Subject :

Within section 2.3.4 of the manual we are able to report the julian day of the location as well as the locations. Within section 2.3.5 we have a whole range of times we can report. I understand from Claudia's email that

she wants to report all of the times associated with time at the surface. We agree and accept all of the surface data and attach the position of the profile to the first resolved position after data start to be received. This may not be the first time data are received. So we write a record in section 2.3.4 for every time a message is received. If there is no position we write fill values. The first time is equivalent to JULD_FIRST_RECEIVE and the last is equivalent to JULD_LAST_RECEIVE. I think Claudia's example shows the same strategy as us. If this is so, what is the advantage of placing these two times also in section 2.3.5? Ann says she would rather things be explicit.
I agree, but balance this aganst the same information appearing twice in the file (and therefore inviting a mistake where they no longer match). My preference is for all of us to do as we and Claudia do now. As Takashi says, unless there is a good and demonstrable reason, let's not add any more things to the format.
TC : ok

From : Claudia Schmid
Date : 10/06/2003
Subject :

Yes, it looks like we are doing the same as Bob Keeley.

Comment: I think the statement `So we write a record in section 2.3.4 for every time a message is received.' is somewhat misleading.

Just for clarification: In our files the times of first and last transmissions are added to the surface position record in consecutive order (and I think Bob is doing the same). For these we take the first/last time stamps of the 32 byte messages (header #2), i.e. we do not use the time of the header line that includes the PTT (header #1) that precedes the first/last transmission (which can be off target by 10 minutes or so).

Something to consider in the decision making process:
Having these two times as part of JULD also helps when estimating the times associated with measurements derived during the drift phase.
TC : ok

From : Annie Wong
Date : 11/06/2003
Subject :

In version 1.0c of the Argo users' manual, p.16,
Section 2.2.5, Calibration info for each profile:

for PARAMETER, please remove PARAMETER:conventions
= "GF3 parameter code", but refer to reference table 3,
TC : ok

so that it is consistent wtih the description for
<PARAM> on p.14, 2.2.4. Measurements for each profile.
Similar edits should also be made elsewhere ... e.g. on
p.33, 2.4.6. Float calibration info., etc.
TC : "GF3 parameter" code is replaced by "Parameter code" in all the user's manual.

From : Annie Wong
Date : 11/06/2003

Subject :

My apologies for not being able to include all comments in one email .... here's another comment for version 1.0c:

for HISTORY, Section 2.2.6, p.18, and Section 2.3.6, p.26, under COMMENT, QCP$ should be "qc performed", not "qc passed". Also, please make reference to Table 11.

TC : ok, "qc passed" is replaced by "qc performed"

From : Bob Keeley
Date : 11/06/2003
Subject :

Thierry, you are quite right about the Provor, they do get reported each cycle. Perhaps all we need in the metadata file is something in the text that says these values are estimates.

From : Bob Keeley
Date : 12/06/2003
Subject :

Annie and I have been working through the QC flags for the scientific QC process and a couple of clarifications need to be sorted out.

When writing QCP$ and QCF$, there has been the debate about storing values as hex or base10. The variable is char*16 and for the moment the maximum value (the sum of all tests) is 7 characters in base10. Some folks seem to have more difficulty dealing in hex than base10. I would be willing to change the specification to base10 from hex if everyone agrees.

We have specified a cdl for the profiles. If we have no conductivity measurements, then what we have written is what is there. If we have, I assume we add a "CNDC" field plus the others as needed into both the cdl and then the data section. Suppose in the history section we have no intention of writing HISTORY_SOFTWARE? Can we just drop this from the cdl? Is this a proper use of netCDF allowing for "flexible" cdl content within a formal structure?

In your example on CNDC, I guess that you are talking about mandatory or not mandatory information in the profile fromat.

In the draft user's manual, there is no mention of mandatory physical parameter.

TC : I think that if a parameter is not measured, then it does not appear in the profile file.

If there is no conductivity measured on a float, then there will be no CNDC, CNDC_QC, CNDC_ADJUSTED, CNDC_ADJUSTED_QC, CNDC_ADJUSTED_ERROR parameters.

There will also be no mention of CNDC in history section.

Did I correctly understand your message on CNDC ?

From : Takashi Yoshida
Date : 19/06/2003
Subject :

I wanted to suggest an editorial correction.

2.5.2 Comment of the PLATFORM_NUMBER

"Q6900045" should be replaced with 6900045.

TC : ok

From : Annie Wong
Date : 19/06/2003
Subject :

Anyhow, while I was going through the history section, I noticed a few more inconsistencies in the Argo User Manual version 1.0c for HISTORY:

on p.17 (profile file) and p.26 (trajectory file),

HISTORY_INSTITUTION:conventions="GTSPP institution code";
HISTORY_ACTION:conventions="GTSPP (MEDS) action code";
HISTORY_PARAMETER:conventions="GF3 parameter code";

These are now obsolete, since all the codes and tables have changed. Also, on p.20, for trajectory files, should
TC : ok
"GF3 parameter code" is replaced by "Argo reference table 3"
"GTSPP (MEDS) action code" is replaced by "Argo reference table 7"
"GTSPP institution code" is replaced by "Argo reference table 4"
"Q where Q =[0-9]" is replaced by "Argo reference table 2"

N_HISTORY = UNLIMITED?
TC : the unlimited dimension for N_HISTORY would be fine. But, there is only one unlimited dimension allowed in a NetCDF file. So N_HISTORY cannot be unlimited.

From : Claudia Schmid
Date : 25/06/2003
Subject :

Dear Charles,

> 1) Is it true that "NO QC WAS PERFORMED" for date and time but
> positions?
>   Most floats have JULD_QC=0 and POSITION_QC=1.

We are doing a speed check, and we check if a date/time is illegal (i.e. year before 1997 or some crazy month, day, ...). What we do not check is if March 31, 2003 is the right date for a profile. The correct date could be March 30, 2003 or April 1, 2003, if the profile was obtained close to midnight (because computer clocks are chronically bad and there are time zone issues, such a test seems not very reliable). That is why we did not assign a 1. However, if it is prefered, we can assign a 1 instead. Please advise.

> 2) Floats 4900271, 4900272, and 4900367 have "JULD_QC=5" and
> "POSITION_QC=0"
> in the latest profile. But, these three floats have "bad" positions (lat
> and
> lon have missing values)

Those are ORBCOMM floats. These floats have `their own way of doing things'. Roger assigns the JULD_QC flag. I think it is possible for them to not transmit a position or date. The date can then be derived in some way (I can ask Roger how he does it), which is why the flag is set to 5.

By the way, we also have some profiles from ARGOS floats without position.

One thing we should change in our files (I think):
JULD_LOCATION should be fill value if LATITUDE and LONGITUDE are fill

value. What do you all think?
TC : is it necessary to write it as a rule ? We may imagine an instrument (ex : GPS) sending a good location time with no latitude or longitude. In that case, latitude and longitude are fill values and juld_location is not a fill value.

Another general thing we may want to change:
I can immagine the crazy case where JULD and JULD_LOCATION may have different flags. Shall we introduce a JULD_LOCATION_QC?
TC : if someone clearly need JULD_LOCATION_QC for a good reason (?) we may introduce it. For the moment, it does not seem necessary.

From : Annie Wong
Date : 23/06/2003
Subject :

Can you please give me two new codes in Reference Table 4 (data centres and institution codes)? I need one for University of Washington, USA, and one for Woods Hole, USA.
TC : Here are the codes you need :

UW	University of Washington, USA
WH	Woods Hole, USA

From : Takashi Yoshida
Date : 14/08/203
Subject :

I would like to ask you to consider wether it is worth introducing an additional variable on float termination time to Argo netcdf files.
Japan Argo recently retrieved four floats by research vessels with the intension of sensor calibration and some technical examinations. In such case, the date (and time) when the float finish its mission is clearly known. I think the information on float termination would be useful. If you try to make a statistical analysis of float life time, those floats which intentionally finish their life time should be eliminated from the statistics, for example.
I think a new variable on float termination time to the metadata file is appropriate. What do you think of it?

TC : the metadata file is the right place for this information.
I propose to create the following entry in the "Float deployment and mission information" section of the meta-data format :

```
char END_MISSION_DATE (DATE_TIME);
END_MISSION_DATE:long_name = "Date (UTC) of the end of mission of the float";
END_MISSION_DATE:conventions = "YYYYMMDDHHMISS";
END_MISSION_DATE:_FillValue = " ";
```

7. Comments on user's manual version 1.0d

From : Thierry Carval
Date : 14/08/2003
Subject : argo user's manual 1.0d

Dear All,

Here is a version 1.0d of the Argo users's manual.
The differences with version 1.0c are typed with a green font.
Most of these differences were minor items. They are explained in the Argo user's manual comments.

I hope that we are now ready to transform this 1.0d proposal into an official 2.0 version of the Argo user's manual.
Your comments are welcome,

From : Takashi Yoshida
Date : 15/08/2003
Subject :

Thank you for the good proposal. In addition to the entry you proposed, another entry which indicates the status of the termination is necessary so that precious information on the way of float termination can be kept. My intention is to distinguish the way of float termination. To the extent I know, there are at least two ways:

1) No more transmission received: In most case, we can recognize the termination by this. The time of the last transmission would be END_MISSION_DATE.
2) Retrieved: If a float is retrieved, the time of retrieval can be known. The float might continue transmission after retrieval. Messages transmitted after retrieval should be regarded as meaningless in both cases of intentional and accidental retrieval.

My proposal of the additional entry is:

```
char END_MISSION_STATUS;
END_MISSION_STATUS:long_name = "Status of the end of mission of the
float";
END_MISSION_STATUS:conventions = "T:No more transmission received,
R:Retrieved";
END_MISSION_STATUS:_FillValue = " ";
```

From : Annie Wong
Date : 27/06/2003
Subject :

It's been a while since I've had to email this group but something popped up this week which need your consensus.

You may remember when we set up the qc flag scale, the purpose is for the delayed-mode loop to correct those data points that have been flagged 1,2,3 by real-time.

While working with a certain PI regarding finalising his delayed-mode data, he expressed that in his way of thinking, after these points have gone through the dm loop and have received their adjustments, they should be considered good to within the supplied error bounds, and hence should all receive a flag of 1.

I like this way of thinking because it puts the emphasis on the error bounds, which is the true nature of dm data (that dm data are an adjustment range but not an absolute value ... the more uncertain the adjustment, the bigger the range). It also makes it obvious that the role of the dm loop is to correct all points that are flagged 1,2,3 in real-time, and turn them into 1 (within the supplied error bounds).

There are, of course, PIs who won't like this black and white view and will still want to use 2,3 to flag their ranges. At the end of the day, it's the PIs choice.

The purpose of this message is not to argue about how people should flag their dm data, but to request an additional wording to qc=1. Currently, the manual reads "1 = Good data: all tests passed". Can we please change it to "1 = Good data: all tests passed; for delayed-mode data, this means good within the supplied error bounds"?

The point is, some PIs will write their dm data this way whether or not we put in this extra qualification, but the users won't know this implicit assumption. So I would rather have the extra bits in the manual than not, even if it only serves the purpose of drawing the user's attention to the error bounds.

TC : ok, here is the definition of code 1 in quality control flag scale (table 2) :
1 : Good data: all tests passed. For delayed-mode data, this means good within the supplied error bounds.

From : Annie Wong
Date : 19/08/2003
Subject :

I picked up some minor typing errors:-

p.9, p.28, p.36, "concentions" should be "conventions".
TC : corrected

p.11, p.21, p.33, delete extra "are" under COMMENT for STATION_PARAMETERS, TRAJECTORY_PARAMETERS, PARAMETER.
TC : corrected

p.26, Arg"o" under DEFINITION for HISTORY_INSTITUTION.
TC : corrected

Other things specific to delayed-mode:-

p.8, "... data to be returned to the global data centres within 3 months." I still don't know what the official timeline is, or who sets this timeline. 3 months is like a promise we can't keep. I think 6 months is more realistic. Maybe Bob and Sylvie can make a decision here.
TC : I write 6

p.14, I would like to see the third paragraph read like this: "Each parameter can be adjusted in delayed-mode. In that case <PARAM> contains the original values, <PARAM>_ADJUSTED contains the adjusted values, <PARAM>_ADJUSTED_QC contains the QC flags set by the delayed-mode process, and <PARAM>_ADJUSTED_ERROR contains the adjustment uncertainties."
TC : ok

p.16, I suggest deleting the following sentences, as they have already been mentioned in previous sections:
"The best scientifically adjusted parameter are stored in the adjusted profile. Example: in a temperature profile (TEMP), the

calibrated values are stored in a adjusted profile (TEMP_ADJUSTED)."
TC : ok

p.23, in the paragraph that starts with "When no parameter is ..." please insert "PARAM_ADJUSTED_ERROR" after PARAM_ADJUSTED_QC and before TRAJECTORY_PARAMETERS.
TC : ok

p.23, the paragraph that starts with "Each parameter can be adjusted ..." Please make it be the same as that on p.14, see comment above.
TC : ok

From : Claudia Schmid
Date : 19/08/2003
Subject :

reference tables are numbered 3.1, 3.2, ... and 1, 2, ...
In the description section they are numbered 1, 2, ...
I think this could be simplified.

Suggestion:
3. REFERENCE TABLES
3.1 TABLE WITH DATA TYPE
3.2 TABLE WITH QUALITY CONTROL FLAGS
And then refer to 3.1, 3.2, ... in the text.
(I can understand if you prefer to leave it as it is, so please don't feel pushed to do it.)
TC : I prefer to separate the number of the table and the number of the chapter describing reference tables (the chapter may change, but not the number of the tables).

p.12: WMO_INST_TYPE does not have an example after `Example :')
TC : corrected

p.15: the parameter definitions have a mistake for all _QC variables: QC:_Fillvalue ="9"; should be QC:_Fillvalue ="0";
TC : corrected

valid_max for pressire: I guess you wanted to enter 12000 not 1200?
TC : I did not find 1200

p.16: three different versions are given for the fill value for calibration date ("0", "", "00000000000000" (I did not count these zeros)). Two of them are in the table and the third is in the example below the table.
TC : ok

p.18, history information example: MEDS needs to be replaced with ME (twice).
TC : ok

p.21 TRAJECTORY_PARAMETERS: ... names are listed in ... (same on p.11, STATION_PARAMETERS; p.33 SENSORS and PARAMETER)
TC : ok

p.23: first line of highlighted text: ... N_PARAM is set to 0; ...
TC : ok

p.24, position accuracy: Thinking of ORBCOMM (positioning system GPS) floats we may have to introduce more possibilities one day. I guess Breck Owens can help with that.
TC : when needed, we shall adapt it.

PARAM_QC and PARAM_ADJUSTED_QC currently has fill_value 9 instead of 0
(the same is the case on p.14).
It might be good to have `<X> this field is specified ...' in
PARAM_ADJUSTED and PARAM_ADJUSTED_ERROR as well (the same on p.14).
TC : ok

p.32: I'm not sure if we need STRING256 for DEPLOY_AVAILABLE_PROFILE_ID.
TC : this is a list of profile_id.

p.33, SENSOR: I think it's not necessary to give the units for the the
three shown sensors in the example. (the same for PARAMETER;
STATION_PARAMETERS on p.11; TRAJECTORY_PARAMETERS on p.21)
TC : ok

p.43 is empty.
TC : ok

p.44, 3.1: the table looks a bit weird because the second column is
basically empty
TC : ok

p.45, 3.4: add NA for NAVO, USA
change 'Woods Hole' to 'Woods Hole Oceanographic Institution' or WHOI (ask
Breck what he prefers)
TC : ok

p.47: 2B+ we also apply visual QC to some profiles, therefore we assign
2B+ without being a GDAC (and I think we and all other DACs have to do
this if we/they do a visual QC).
Note 3: 'Even though the data at Global Data Centres go through
semi-automatic or manual QC ... intercomparison ...'
TC : ok

p.50 'The Pacific/Atlantic boundary is complicated' ;-)
I suggest to replace frontier with boundary.
TC : ok

p.52: which web site? (needs to be completed)
TC : ok

Now I have a question about the treatment of trajectory files when delayed
mode data come in. We will merge the real-time with the delayed mode data,
so as to always distribute the best data possible. This makes it hard to
chose an appropriate data state indicator. I think the best is to always
assign the one used for real-time data. What do you think?
TC : if the trajectory data are not corrected, we do not need to assign a
delayed mode data state indicator.

This is all I found. Now I pass my copy on to Yeun-Ho.

From : Annie Wong
Date : 14/10/2003
Subject :

In view of recent developments in delayed-mode procedures,
I think for User Manual Version 2, in Reference Table 11:
QC Test Ids,

```
131072 = Wong et al correction

should now be

131072 = delayed-mode statistical test
TC : ok
```

From : Rebecca McCreadie
Date : 08/10/2003
Subject :

```
In the trajectory files I think we need another flag defined for
JULD_ASCENT_START_STATUS
JULD_ASCENT_END_STATUS
JULD_DESCENT_START_STATUS
JULD_DESCENT_END_STATUS
JULD_START_TRANSMISSION_STATUS

At the moment there isn't a defined flag for when you have had to fill
the associated parameter. Could I suggest 9: date is unknown
9 is in-line with the other flags used.
```

TC : ok

From : Rebecca McCreadie
Date : 03/10/2003
Subject :

In history information section, a HISTORY_REFERENCE_DATA_BASE item should be inserted. It contains the reference data base of the institution.

From : Annie Wong
Date : 03/10/2003
Subject :

Two columns should ne added to the QC flag reference table (2). One for explanations on real-time qc, a second for delayed-mode qc.

From :
Date :
Subject :

From :
Date :
Subject :